

OTS PRICE

\$ 8.60 per  
copy

XEROX

\$ 2.99 per

MICROFILM

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION  
CONTRACT NO. NAS7-100

Technical Memorandum No. 32-75

**Stiffness Matrix Structural Analysis**

R. R. Bachelder  
R. K. Wade

  
C. E. G. Cheever  
Chief  
Professional Services

JET PROPULSION LABORATORY  
CALIFORNIA INSTITUTE OF TECHNOLOGY  
PASADENA, CALIFORNIA

February 12, 1962

## CONTENTS

<b>I. Introduction</b>	1
A. General Description	1
B. Function of Program	1
C. Method of Analysis	4
D. Operating Experience	3
<b>II. Mathematics</b>	5
A. Notation	5
B. Derivation of Matrix Equations	5
C. Generation of the Stiffness Matrix	4
D. Generation of Weight and Load Matrices	6
E. Static Analysis	6
F. Normal-Mode Analysis	6
G. Accuracy	10
H. Structural Idealization	13
<b>III. Programming</b>	14
A. Input Format	14
B. Output Format	16
C. Operator Instructions	17
D. Flow Charts	17
E. Program Listing	55
<b>Appendix A. Matrices for Various Member Types</b>	73
<b>Appendix B. Stresses for Various Member Types</b>	79
<b>Appendix C. Example Problem</b>	81
<b>Bibliography</b>	89

Copyright © 1962  
 Jet Propulsion Laboratory  
 California Institute of Technology

## I. INTRODUCTION

### A. General Description

A family of programs has been developed at JPL to analyze structural frameworks. Since the programs are intended for use as a design tool, particular attention has been given to simplicity and flexibility of input and output. They may thus be used by personnel who have had little training in computer utilization, and input may easily be revised to reflect changes in a design.

The programs are coded in FORTRAN language, and may be run at any IBM 704-7090 installation whose system is compatible with that at the Jet Propulsion Laboratory and whose machine has a 32K memory.

Four similar programs have been written for the analysis of four types of structure:

1. Three-dimensional structure, pinned joints
2. Three-dimensional structure, rigid joints, circular member cross sections
3. Planar structure, rigid joints, loaded in-plane
4. Planar grid structure, rigid joints, loaded normal, in-plane

### B. Function of Program

A structural framework will be defined as a stable system of uniform, weightless members, and joints at which loads are applied and inertias are lumped. Such a framework and its environment may be described by the following quantities:

1. Coordinates of joints
2. Geometric and elastic properties of members
3. Locations of restraints
4. Inertias at joints
5. Static loads at joints
6. Acceleration of a joint during free vibration in a normal mode

Given these as input, the program will perform the computations to provide as output:

1. Deflections and member stresses for static loadings
2. Frequencies, mode shapes, and member stresses during free vibration in normal modes

**C. Method of Analysis**  
 Each program generates the stiffness matrix  $K$  for a particular type of structure from geometrical data, and performs static and normal-mode analyses by solving the equations

$$U = K^{-1} F \text{ and } \frac{1}{\omega^2} U = K^{-1} M U;$$

where  $F$  is a matrix of static loads,

$M$  is a matrix of inertia terms,

$U$  is a matrix of static deflections or a normal-mode shape,

and  $\omega$  is the circular frequency of a normal mode.

Member stresses are computed from a set of deflections  $U$  and the geometrical properties of the member.

The stiffness matrix method of analysis was chosen over possible techniques (e.g., flexibility matrix, force relaxation) because it more fully satisfies the following criteria:

1. That it provide a complete analysis (deflections, stresses, normal modes)
2. That the input be in a simple form
3. That it analyze statically indeterminate structures with no extra effort on the part of the user
4. That it be adaptable to any type of framework
5. That a useful program be easy to write

6. That the computer be utilized efficiently with respect to storage capacity and running time
7. That the accuracy of the solution be sufficient for engineering use and be predictable

**D. Operating Experience**

The program has been used extensively during design of the Mariner A vehicle. In the few cases where prototype experimental data are available, correlation with predicted results is good. Analyses of structures of 110 degrees of freedom have been performed with no accuracy problem, as indicated by a check on static equilibrium of the structure.

Machine time on the 7090 for complete analyses (static and normal mode) varies from one minute for 20 degrees of freedom to 30 minutes for 130 degrees of freedom.

The input for the original Mariner A basic structure, as an example, could be written in about two hours after appropriate idealization. (The structure was of 90 degrees of freedom, statically indeterminate to the 4th degree.) Key-punching the data cards required 20 minutes; machine time was about 10 minutes. More than 25 revisions to the original data have been run during the design process.

Some experimentation has been done with very poorly conditioned matrices (in particular, Hilbert matrices) to determine the effect of conditioning on accuracy. Empirical results of these tests are presented in Section II-G.

## B. MATHEMATICS

### A. Notation

$A$	member area; square matrix	$U$	matrix of static deflections
$A_1$	input member section property	$U_n$	vector mode shape, nth normal mode
$a_i$	acceleration of structure in $x_i^1$ direction	$s_{ij}$	deflection in $i$ th generalized component direction
$a_{ij}$	element of matrix $A$	$s_{ii}$	magnitude of $s_{ij}$ in the $i$ th normal mode
$B$	member bending rigidity; square matrix	$t_{ij}$	deflection of joint $p$ in $x_i^1$ direction
$C$	member section parameter; square matrix	$v_p$	vector deflection of joint $p$ in $x_i^1$ component
$c_{ij}$	element of matrix $C$	$V_n$	eigenvector, nth mode; shear stress at joint in
$D$	outside diameter of circular member section	$W$	width of rectangle; near our section
$E$	elastic modulus	$W_i$	component of weight (or weight moment of inertia) in $x_i^1$ direction
$F$	matrix of static loadings	$X_n^{(1)}$	trial vector, $n$ th mode, $1$ th iteration
$f_i$	load in $i$ th generalized component direction	$x_{ij}^{(1)}$	$i$ th component of $X_n^{(1)}$
$f_{ij}$	load at joint $p$ in the $x_i^1$ direction	$z_{ij}$	coordinate of joint $p$ in $x_i^1$ direction
$f_p$	vector load applied to joint $p$ in $x_i^1$ components	$z_i$	unit vector in $x_i^1$ , coordinate direction
$f_n$	natural frequency in $n$ th mode, cps	$a_n$	constant
$g$	gravity acceleration, in./sec. <sup>2</sup>	$\gamma_i$	cosine of angle between member axis and $x_i^1$ axis
$H$	depth of rectangular member sections	$\delta_i$	vector deflection in $x_i^1$ direction
$I$	moment of inertia of member sections; unit metric	$c$	constant
$I$	input joint number	$\lambda_n$	eigenvalues, $n$ th mode
$K$	member section horizontal stiffness parameter; square stiffness matrix	$P$	Poisson's ratio
$k_{ij}$	element of matrix $K$	$\rho$	unit vector in $i$ th coordinate direction of member-oriented coordinate system
$M$	diagonal matrix of inertial terms; bending moment	$\epsilon_{ij}$	cosine of angle between $x_i^1$ and $x_j^1$ coordinate axes
$m_i$	inertia in $i$ th generalized component direction	$m_c$	circular frequency, rad/sec
$m_{pq}$	inertia at joint $p$ in $x_i^1$ direction	<b>Sign Conventions</b>	
$N_i$	input control parameter	1. Right-handed coordinate systems	
$n$	degree of freedom of structure	2. Forces and displacements positive in positive coordinate directions	
$P$	condition number; axial stress	3. Moments and rotations positive by right-hand rule about positive coordinate axes	
$p_i$	constant		
$r_i$	input retractor parameter		
$s_n$	acceleration in $n$ th mode, in./sec. <sup>2</sup>		
$S$	member length		
$T$	wall thickness of circular member section		

**B. Derivation of Matrix Equations**

At any joint in a structure, a component of load  $f_i$  applied to the joint must be equilibrated with member stresses acting on the joint in the same direction. Since

member elements in a linear structure are proportional to deflections. If the equations of force equilibrium in all degrees of freedom system may be written

$$k = \sum_{i=1}^n k_{ij} p_i, \quad i = 1, n, \quad (1)$$

where the  $k_{ij}$  are constants of proportionality. In matrix notation, the same equation is

$$p = KU.$$

When a joint undergoes free vibration in a normal mode  $n$ , the component deflections must be of the form

$$u_i = f_i \sin \omega_n t. \quad (2)$$

The member load acting in the same direction is

$$f_i = -m_i \omega_n^2 u_i = m_i \omega_n^2 f_i \sin \omega_n t. \quad (3)$$

Substituting this load into the expression for force equilibrium

$$m_i \omega_n^2 u_i + \sum_{j=1}^n k_{ij} p_j = 1, n, \quad (4)$$

or, in matrix notation,

$$M U_n + p_n = K U_n.$$

The analysis thus involves solution of two matrix equations: knowing a set of loads  $p$  to compute static displacement  $U$  from

$$p = KU;$$

and knowing the matrix of the structure  $M$  to compute normal-mode shapes and frequencies (eigenvalues and eigenvectors)  $U_n$  and  $\omega_n^2$  from

$$M U_n + p_n = K U_n.$$

Member stresses may be computed from static displacements  $U$  or properly normalized mode shapes  $U_n$

$$F = KU;$$

and knowing the inertia of the structure  $M$  to compute normal-mode shapes and frequencies (eigenvalues and eigenvectors)  $U_n$  and  $\omega_n^2$  from

$$M U_n + p_n = K U_n.$$

The coefficient  $k_{ij}$  is then the force requirement in the  $i$ th direction per unit deflection in the  $j$ th direction, all other directions being zero.

The matrix of coefficients  $k_{ij}$  for a member of any type connecting joint  $p$  and  $q$  is derived by introducing unit component deflections of  $p$  and  $q$ , and calculating the forces at  $p$  and  $q$  produced by each deflection. Matrices for several types of members are presented in Appendix A.

To illustrate the method by which such matrices are computed, and how they are used in the generation of a matrix for a structure, consider the pin-jointed member in two dimensions as shown in Sketch 1:



The solution of this equation may be further substantiated by writing

$$\begin{bmatrix} f_p \\ f_q \\ f_r \end{bmatrix} = \begin{bmatrix} K_{pp} & K_{pq} \\ K_{qp} & K_{qq} \end{bmatrix} \begin{bmatrix} p \\ q \end{bmatrix},$$

where the vectors have components

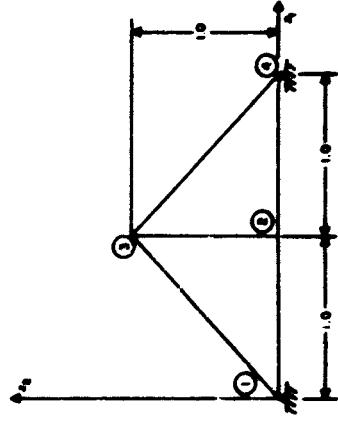
$$\begin{bmatrix} f_p \\ f_q \\ f_r \end{bmatrix} = \begin{bmatrix} f_{1x} \\ f_{1y} \\ f_{2x} \\ f_{2y} \end{bmatrix}, \quad \begin{bmatrix} p \\ q \end{bmatrix} = \begin{bmatrix} u_{1x} \\ u_{1y} \\ u_{2x} \\ u_{2y} \end{bmatrix}, \quad \text{etc.}$$

and the elements of  $K_{pq}$  are products of the three columns of the stiffness matrix. These forces, being reacted by springs in the members in point 1, are denoted by the members of members 1,2 and 1,3 as follows:

$$\begin{bmatrix} f_p \\ f_q \\ f_r \end{bmatrix} = \begin{bmatrix} K_{11} & K_{12} & K_{13} \\ K_{21} & K_{22} & K_{23} \\ K_{31} & K_{32} & K_{33} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} = 10^6 \begin{bmatrix} p \\ q \\ r \end{bmatrix}. \quad (4)$$

4. The stiffness matrix is formed by similar steps:

$$\begin{bmatrix} f_p \\ f_q \\ f_r \end{bmatrix} = \begin{bmatrix} K_{11} & K_{12} & K_{13} \\ K_{21} & K_{22} & K_{23} \\ K_{31} & K_{32} & K_{33} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} = 10^6 \begin{bmatrix} p \\ q \\ r \end{bmatrix}. \quad (5)$$



$$\frac{\partial f}{\partial u} = 10^6 \cdot 10^3 \cdot \text{mm}^3 \cdot \text{N/mm}^2. \quad (\text{mm}^3 \cdot \text{N/mm}^2 = \text{kg/mm}^2)$$

$$f_{11} = -\frac{4B}{3} \gamma_1$$

$$f_{21} = \frac{4B}{3} \gamma_1 \gamma_2$$

$$f_{31} = -\frac{4B}{3} \gamma_2$$

$$f_{41} = \frac{4B}{3} \gamma_2$$

$$f_{51} = -\frac{4B}{3} \gamma_2$$

$$f_{61} = \frac{4B}{3} \gamma_2$$

5. Matrices are introduced in the form  $S_{ij} = S_{ji} = u_{ij} = u_{ji}$  through the matrix, the first, second, and last columns make no contribution to the product and may be omitted from the operations. Also, the forces  $f_{ij} / f_{ji} = u_{ij}$  are unknowns required to be determined from the unknown deflection components.

6. Analysis for the unknown deflections thus reduces to solution of the equation

$$\begin{bmatrix} f_{11} \\ f_{21} \\ f_{31} \\ f_{41} \\ f_{51} \\ f_{61} \end{bmatrix} = 10^6 \begin{bmatrix} 2 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -0.5 \\ 0 & -1 & 0 & 2 & 0.5 \\ -1 & 0 & -0.5 & 0.5 & 1.5 \end{bmatrix} \begin{bmatrix} u_{11} \\ u_{21} \\ u_{31} \\ u_{41} \\ u_{51} \end{bmatrix}.$$

In summary, the program generates the stiffness matrix as follows:

1. Step through joints consecutively. For joint  $i$ :
2. Search list of member numbers for  $p$ . For each member  $p$ :
3. Compute and store (temporarily) the matrix entries corresponding to deflection  $u_{ip}$ . The submatrix  $A_{ip}$  has the following locations:

$$\begin{bmatrix} f_{11} \\ f_{21} \\ f_{31} \\ f_{41} \\ f_{51} \\ f_{61} \end{bmatrix} = \begin{bmatrix} K_{11} & \{u_i\} \\ \vdots & \vdots \\ K_{21} & \{u_i\} \\ \vdots & \vdots \\ K_{31} & \{u_i\} \\ \vdots & \vdots \\ K_{41} & \{u_i\} \\ \vdots & \vdots \\ K_{51} & \{u_i\} \\ \vdots & \vdots \\ K_{61} & \{u_i\} \end{bmatrix}.$$

4. Search list of component matrices, delete rows and combine stiffness matrix vertically.

5. Move unselected columns into static stiffness matrix entry, except where a column corresponds to a zero deflection component as determined by checking the list of component numbers.
6. A few properties of the stiffness matrix are evident from its definition:

1. It is symmetric, a consequence of Marvel's reciprocity theorem.

2. Each diagonal term is positive and is large compared to all other elements in its row, since diagonal blocks are formed by superposition of off-diagonal blocks.

3. Stability of the structure is reflected in the linear independence of  $\{u_i\}$ , after rows have been deleted to account for restraints.

4. It is generally sparse (many elements are zero), since the position of matrix elements reflects the presence of members.

#### D. Generation of Weight and Load Matrices

The matrices  $M$  and  $F$  are generated by appropriate storage of input quantities and converted to account for restraints in the same manner as for stiffness matrix entries. The diagonal matrix  $M$  is stored as a vector.

Loads may be specified either as concentrated point forces or moments, or as linear accelerations of the structure as a rigid body. Loads at joint  $p$  corresponding to acceleration in the  $i$ th coordinate direction  $a_i$  are computed as

$$f_{pi} = m_{pi} a_i.$$

#### E. Static Analysis

Given the matrices  $K$  and  $F$ , the deflections  $\{u\}$  are computed from

$$U = K^{-1} F$$

by Gaussian elimination. No row interchanges or pivot tests are performed, since the diagonal of the stiffness matrix is always strong, i.e., the diagonal element is the largest number in its row. Overdetermined clustering an arithmetic operation is not needed, so the elimination process continues until whatever remains in the accumulation.

Experience with this basic procedure has been good. It has provided results in highly ill-conditioned problems which compare favorably with those computed by more sophisticated techniques.

Static member stresses are computed from the deflections  $U$  and the geometry of the structure. Appropriate equations for each member type are given in Appendix B.

#### F. Natural Mode Analysis

An iterative procedure for computing solutions  $U_n$  and  $V_n$  of the equation

$$K U_n = \omega_n^2 M U_n \quad (3)$$

is developed in this section.

First, the above equation will be transformed into

$$A V_n = \omega_n^2 V_n + \sigma_n \quad (4)$$

where  $A$  is real and symmetric. Solutions to this equation have the following properties (Ref. 1-6):

1. There are  $n$  solutions  $\{u_n\}$ ,  $V_n$ , where  $A$  is of order  $n \times n$ .
2. The eigenvalues  $\omega_n^2$  are all real and positive, and the eigenvectors  $V_n$  are real.
3. The eigenvectors are orthogonal with respect to the unit matrix, i.e.,

$$V_1^T V_m = 0 \quad k \neq m. \quad (5)$$

4. The length of an eigenvector is indeterminate; i.e., if  $V_n$  is a solution,  $c_n V_n$  is also a solution, where  $c_n$  is a constant.
5. Any vector  $X$  of order  $n$  may be represented by a linear combination of eigenvectors; i.e.,

$$X = \sum_{n=1}^n a_n V_n. \quad (6)$$

There are two principal reasons for performing the transformation:

1. Equation (3), representing an undamped structure, can only have real, positive eigenvalues. It is possible, however, for roundoff during operations on  $K$  and  $M$  to produce an equation of similar form with imaginary components in its solution. The convergent process, using real arithmetic, will not converge on such solutions. This problem is avoided if the matrix  $A$  in Eq. (4) is kept symmetric.
2. Use of the orthogonality condition is simpler if eigenvectors are orthogonal with respect to the unit matrix rather than to another matrix ( $\omega_n^2 V_1^T M V_n = 0$ ).

The transformation is effected as follows: Define

$$A = M^{-1/2} A M^{-1/2} \quad (7)$$

where, since  $M$  is a diagonal matrix of positive elements,  $M^{-1/2}$  is also a diagonal matrix whose element in the  $i$ th row is  $1/\sqrt{m_i}$ , and the corresponding element in  $M^{-1/2}$  is  $1/\sqrt{m_i}$ . Also, let

$$V_n = M^{1/2} U_n. \quad (8)$$

and similarly,

$$Substituting Eq. (7) into Eq. (3),$$

$$K U_n = \omega_n^2 M U_n. \quad (9)$$

Substituting Eq. (8) into Eq. (9),

$$M^{-1/2} K M^{-1/2} U_n = \omega_n^2 U_n. \quad (10)$$

Since  $K$  is symmetric, the product

$$A = M^{-1/2} K M^{-1/2}$$

is symmetric and the desired formulation

$$A V_n = \omega_n^2 V_n$$

is achieved, where

$$U_n = M^{-1/2} V_n.$$

and  $u_n$  are the desired solution.

The solutions of Eq. (3) corresponding to smallest values of  $\omega_n$  are of primary importance in structural applications, since larger deflections and strains occur during vibration at lower frequencies. The iterative process to be described converges most readily on the eigenvalue of largest magnitude, so a transformation of Eq. (4) is performed:

$$C V_n = \lambda_n V_n. \quad (10)$$

where  $C = A^{-1}$  and  $\lambda_n = 1/\omega_n^2$ .

The inverse is computed by straightforward gaussian elimination on the upper triangular half. No row interchanges or checks for division by zero pivot elements are performed.

Solutions of Eq. (10) for the largest value of  $\lambda_n$  and the corresponding value of  $V_n$  will now be found. From Eq. (6) any vector

$$X = \sum_{n=1}^n a_n V_n,$$

$$C X = C \left( \sum_{n=1}^n a_n V_n \right)$$

$$= \sum_{n=1}^n a_n C V_n$$

$$= \sum_{n=1}^n a_n \lambda_n V_n;$$

$$C(CX) = CX - C\left(\sum_{n=1}^{\infty} \alpha_n V_n\right)$$

$$= \sum_{n=1}^{\infty} \alpha_n A V_n.$$

In general,

$$C^n X = \sum_{n=1}^{\infty} \alpha_n A^n V_n. \quad (11)$$

If the multiplication process is continued, the right side of Eq. (11) will eventually be dominated by powers of the largest eigenvalue  $\lambda_1$ :

$$C^n X \rightarrow \alpha_1 A^n V_1, \quad k \rightarrow \infty.$$

In practice, to keep the components of  $X^{(1)}, \dots, X^{(k)}$  from becoming too large,  $X^{(1)}, \dots, X^{(k)}$  is normalized after each multiplication so that its largest component is 1. (This is equivalent to scaling the lengths of the  $V_n$ 's by an arbitrary factor.) The procedure described above will be modified if the trial vector becomes too large.  $X^{(1)}, \dots, X^{(k)}$  is normalized until  $X^{(k)}$  converges to  $V_1$ , and the normalization factor is  $\alpha_1$ . Multiplication continues until the maximum difference between components of  $X^{(1)}, \dots, X^{(k)}$  and  $X^{(k+1)}$  is within a given tolerance, or until a maximum number of cycles has been performed.

For obvious reasons, the foregoing procedure is called "the power method." It is a generalization of Steklov's method where successive powers of a single shape  $X^{(1)}$  are used to compute better guesses:

$$CX^{(1)} = \alpha_1 X^{(1)}, \text{ etc.}$$

At any stage in the convergent process, an approximate eigenvalue of better accuracy than the current eigenvalue is given (Ref. 1, p. 83) by Rayleigh's Quotient, defined as

$$\lambda_1 = \frac{\lambda_1^T CX}{\lambda_1^T X}.$$

II. In Eq. (8),  $\alpha_1 = 0$ ,

$$X = \sum_{n=1}^{\infty} \alpha_n V_n,$$

$$CX = \sum_{n=1}^{\infty} \alpha_n A V_n,$$

then convergence will be to the next best eigenvalue  $\lambda_2$  and eigenvector  $V_2$ . The condition may be obtained by application of the orthogonality condition of Eq. (5) to keep an arbitrary vector  $X$  orthogonal to  $V_1$ , (or any

lower eigenvectors). Thus, if  $V_1$  is known, the transformation of an arbitrary vector  $X$  to a vector  $X'$ , orthogonal to  $V_1$ , is as follows:

$$X = \sum_{n=1}^{\infty} \alpha_n V_n.$$

$$V_1^T X = \sum_{n=1}^{\infty} \alpha_n V_1^T V_n = \alpha_1 V_1^T V_1,$$

$$V_1^T X' = 0 = V_1^T X - \alpha_1 V_1^T V_1,$$

$$X' = X - \alpha_1 V_1$$

$$= X - \frac{V_1^T X}{V_1^T V_1} V_1.$$

Similar transformations orthogonalize  $X$  to other eigenvectors  $V_2, V_3$ , etc.

When eigenvalues are close, say

$$\lambda_1 \approx \lambda_2;$$

the trial vector becomes

$$X^{(1)} = \alpha_1 A^1 V_1 + \alpha_2 A^1 V_2,$$

in which powers of  $A$  cannot dominate those of  $\alpha$  for any reasonable  $k$ . The process described above will be modified to speed convergence to the larger of close eigenvalues. As before,

$$X = \sum_{n=1}^{\infty} \alpha_n V_n.$$

Given an arbitrary number  $p$ ,

$$(C - \rho I) X = \sum_{n=1}^{\infty} \alpha_n (C - \rho I) V_n$$

$$= \sum_{n=1}^{\infty} \alpha_n (CV_n - \rho V_n)$$

$$= \sum_{n=1}^{\infty} \alpha_n (\lambda_n - \rho) V_n.$$

Variations of  $p$  with the present method may be easily described by the continual product relation

$$(C - \rho I)^k X = \sum_{n=1}^{\infty} \alpha_n (\lambda_n - \rho)^k V_n,$$

which converges to

$$(C - \rho I)^k X \rightarrow \alpha_1 (\lambda_1 - \rho)^k V_1, \quad k \rightarrow \infty,$$

where  $(\lambda_1 - \rho)^k$  is the largest value of the differences. The problem now is to obtain values of  $p$  which

which eliminate products of  $(C - \rho I)$  and  $(\lambda_n - \rho)$  with  $p$  varying from  $p_1$  to  $p_2$ .

The procedure for automatic selection of  $p$  in Eq. (11) may be summarized as follows:

1. Set  $p = 0$ . (No initial estimate of the highest eigenvalue by five iterations on  $\alpha_1$ .)
2. Alternate  $p = \lambda_{n+1}, 0, \dots, \lambda_1, 0, \dots, \lambda_{n+1}$ . If  $\lambda_{n+1} < 0.01$ , set  $p = 0$  to force convergence on  $\lambda_n - \rho$ . If  $\lambda_{n+1} > 0.001$ , use  $p_1 = 0.001$  to prevent machine underflow in the desired eigenvector.

3. Very  $p$  in the range  $0 \leq p \leq 0.015$  to a quadratic formula emphasizing values of  $p$  near zero. Repeat a maximum of 50 steps for each mode, checking convergence at each cycle.
4. Repeat steps 3 and 3 five times.

When two (or more) eigenvalues are equal,

(then, after many iterations the trial vector  $X = \alpha_1 V_1 + \alpha_2 V_2$ ),

where the  $\alpha$ 's are arbitrary; thus, there can be no convergence to a "true" eigenvalue although the eigenvalue  $\lambda_1 - \rho$  is well defined. Consider, for example, a system of three equations with a rigid stability and one of weightless constraint which is rigid stability and of circular cross section. The position of the mass has two coordinate components to the system due to designation of freedom, two mode shapes, and two equal frequencies. Using the power method, there would be no convergence to a first mode shape, since this could be done in any direction. When iterations are stopped, however, convergence on a second mode (orthogonal to the first) will be obtained. The same is true in the case of many degrees of freedom and several identical frequencies.

There is still the possibility that convergence on the first eigenvalue and

will not be refined enough to eliminate components of lower vectors in  $X$ . No test on this error is available. In practice, enough iterations have been made that probably reasonable mode shapes have been obtained in every problem with multiple eigenvalues.

Convergence is tested by searching for the minimum difference between elements in successive trial eigenvalues. If all

$|x_{i+1}^{(k)} - x_{i+1}^{(k-1)}| \leq \epsilon, i = 1, n$ ,  
no the amplitude of acceleration in the  $i$ th generalized direction is

$$\theta_{i+1} = \theta_{i+1} + \epsilon.$$

Directions are stepped on that mode and begun on the next. The creates a vector from  $4 \times 10^3$ , when course elements of the eigenmode are required to  $4 \times 10^3$  for the final option.

Initial guesses at the trial vectors  $X_0^{(k)}$  are required to start convergence on each of the six modes computed. These are taken as successive normalized products of the diagonal of  $C \times C$ , in reverse order:

$$x_0^{(k)} = e_{i+1}$$

$$X_0^{(k)} = C^{(k-1)} X_0^{(k)}$$

The vector  $X_0$  should be a fair guess of the first mode shape, since its component parts are largest where mass and flexibility are largest. This guess improves with successive iterations, so  $X_0$  may be closer to  $V_0$ , before convergence is tested. Higher mode guesses  $X_0$  are similarly affected by mass and flexibility, so when they are unchanged in lower modes they may be expected to converge relatively rapidly as well. Experience with this procedure has been satisfactory.

In theory, it is possible to compute lower frequencies from the original Eq. (4) of the problem

$$\Delta F_{i+1} = -\omega_i^2 V_{i+1}$$

by choosing  $p > 1$ , so that  $(\omega_i^2 - p)$  has a larger magnitude than any other  $(\omega_j^2 - p)$ . This has the advantage that computation of  $C = A^{-1}$ , with attendant errors, is eliminated. In practice, however, the lower eigenvalues are so close in comparison with the upper ones that convergence is probably slow. Also, there is evidence that eigenmodes computed by this process are more in error than those obtained from even a poor inverse. Similarly, although the accuracy of edge vectors computed from

$$C V_0 = A_0 V_0$$

is dependent on the accuracy of the inverse  $C = A^{-1}$ , the eigenmodes will not be improved by iteration through

$$\Delta V_0 = -\omega_0^2 V_0.$$

If the acceleration is a component direction of a joint in known when a structure is undergoing vibration in a normal mode, the absolute magnitudes of the mode shape is determined and stresses may be computed. An acceleration may be known from previous dynamic testing, or analysis of an simulated damaged version of the structure. Definition one of the pertinent form

$$\alpha_0 = g \sin \theta_{i+1},$$

5. Failure to properly test convergence of the normal-mode analysis.

6. Gross program or machine errors. The programs have been tested on check problems and on nearly 100 production runs. Correlation with test results has been good where tests have been run. The machine usually detects its own errors, prints a diagnostic, and stops; the problem must then be rerun.

Single-precision arithmetic is used throughout; this provides storage of approximately eight decimal digits plus exponent for all quantities. Required accuracy for engineering purposes is two or more significant figures for the largest quantities in a set of calculations.

Input  $w_{i+1}^{(k)}$  usually be provide-1 with three or more significant digits, with some filling out the stored number of eight digits. Computation during matrix generation introduces roundoff in the last one or two places of the elements of the stiffness matrix.

Accuracy of the static analysis

$$U = K^{-1} P$$

is impaired if the stiffness matrix is singular or ill-conditioned. Indeed, Singularity is caused by structural instability and causes division by zero, since no overflow checks are made to detect division by zero. The only indications are those cited below for ill-conditioning. This letter is a qualitative description of the loss of accuracy during computation of an inverse. Generally, significant figures are lost during subtraction operations when digits ( $r$ ) subject to roundoff are drawn into the significant places of a number, e.g.:

$$\frac{0.1234567 \times 10^6}{0.000001 \times 10^6} = 0.1234567 \times 10^6$$

It has been observed in structural usage that ill-conditioning becomes a problem when the stiffnesses of joints are greatly different. Thus, when the ratio of diagonal elements of  $K$  were

$$\frac{\lambda_{11}}{\lambda_{11}} < 10,$$

systems of 100 degrees of freedom were successfully solved, while smaller systems with

$$\frac{\lambda_{11}}{\lambda_{11}} > 100$$

gave obviously false results. A second indication of ill-conditioning is the ratio of maximum to minimum eigenvalues of  $K$ , or condition number

$$P = \frac{\lambda_{max}}{\lambda_{min}},$$

This number is computed for

$$A = M^{-1} K M^{-1},$$

and only indicates the condition of  $K$  when the elements of  $M$  are nearly equal. Condition numbers  $P > 10^6$  generally indicate loss of all significance from computer calculations.

Normal-mode analysis is subject to the same problems of singularity and ill-conditioning as static analysis, plus problems caused by the nature of  $M$ , and the convergent means of solution. If  $M$  contains zero diagonal elements, then  $M^{-1}$  will have elements produced by division by zero and

prove to be singular. If the ratio of elements of  $M$  are large, so that ratio of diagonal elements of  $A$  are large,

$$\frac{\lambda_{11}}{\lambda_{11}} > 100,$$

then  $A$  may be ill-conditioned and

$$C = A^{-1}$$

subject to large error.

Convergence is tested by comparing elements of normalized trial vectors at successive iterations

$$C X_0^{(k)} = e_0 X_0^{(k+1)},$$

when

$$|x_{i+1}^{(k)} - x_{i+1}^{(k+1)}| \leq \epsilon,$$

Iterations are stopped. Alternative tests are

$$(C - A^{(k)}) V_0^{(k)} = X_0^{(k)}$$

$$|x_{i+1}^{(k)}| \leq \epsilon,$$

which may never stop iteration, and

$$|\lambda_{11}^{(k)} - \lambda_{11}^{(k+1)}| \leq \epsilon,$$

which proves to stop the convergent process too soon. There is always a risk that convergence will stop too soon when it is very slow, since the change in any parameter then becomes small even when the parameter is far from its true value. Checks against this possibility include model testing and computation of normal modes by an

iteration method, e.g., Jacobi's method. The maximum change in a vector element is output for checking; this value should be

$$\epsilon \leq 4 \times 10^{-7}$$

at the final iteration of each mode.

The ratio of maximum to minimum eigenvalues of a matrix or condition number is a measure of the degree of ill-conditioning of the matrix. The maximum eigenvalue of

$$AV = \lambda_0 V,$$

$\lambda_0^2$  may usually be found easily and accurately by the power method. The minimum eigenvalue,  $\lambda_1$ , is found from

$$CV_n = \lambda_1 V_n,$$

where

$$C = A^{-1}.$$

As noted before, condition numbers

$$P = \frac{\lambda_0^2}{\lambda_1^2} < 10^6$$

usually indicate that engineering accuracy can be obtained.

A consequence of the power method is that eigenvalues are computed in descending order. If such is not the case, there probably has been no reliable convergence to one or more eigenvectors. Since frequencies are proportional to reciprocals of eigenvalues of  $C = A^{-1}$ , the output frequencies must be in ascending order. (When frequencies are very close—differing in the third place—this rule may be violated without prejudicing the results.)

The following eigenvalues are defined as:

$\lambda_0$  = computed lower eigenvalue of  $A$ .

$\lambda_0^2$  = corresponding computed upper eigenvalue of  $C = A^{-1}$ .

$\lambda_0$  = true magnitude of lower eigenvalue of  $A$ .

It has been observed in tests with Hilbert matrices that the difference

$$\left| \lambda_0 - \frac{1}{\lambda_0^2} \right| > \left| \lambda_0 - \frac{1}{\lambda_0} \right|$$

Also, in all reliable production runs,

$$\lambda_0 \lambda_0^2 \approx 1.$$

Thus, if

$$|\lambda_0 \lambda_0^2 - 1| > 10^{-4},$$

the validity of  $\lambda_0$  and its eigenvector should be doubted; and, if not, then the error in the computed eigenvalue is

$$\left| \lambda_0 - \frac{1}{\lambda_0^2} \right| < \left| \lambda_0 - \frac{1}{\lambda_0} \right|.$$

In summary, the following checks are available in the user program output:

1. Normal-mode convergence test,  $\epsilon < 4 \times 10^{-7}$
2. Condition number,  $P < 10^6$

3. Frequency output in ascending order unless nearly equal.
4. Eigenvalues of  $C$  regular eigenvalues of  $A$  within

$$|\lambda_0 \lambda_0^2 - 1| < 10^{-4}.$$

The following test may be applied as the need arises:

1. Hand-check of program input
2. Readability of program output
3. Static equilibrium of loads and reactions
4. Ratios of stiffness matrix diagonal elements

$$k_{ii}/k_{jj} < 100$$

5. Solution by independent numerical methods

6. Comparison with different idealizations of the same configuration
7. Model testing

#### H. Structured Idealizations

Matrix representations of four distinct types of structures have been programmed. Any structure to be analyzed must be idealized by a structure composed entirely of members of one of these types:

1. Three-dimensional, pin-jointed
2. Three-dimensional, rigid-jointed, with circular members, cross sections
3. Planar, rigid-jointed, loaded in-plane
4. Planar, rigid-jointed, loaded normal-to-plane (grid)

Some types of idealization are commonly used in structural analysis; for example, trusses are usually assumed to be pin-jointed, and continuous slabs are often analyzed as grids. The following remarks will be concerned with typical approximations which extend the power of the programs:

1. A continuous structure may be approximated by a "hunged-mass" system. Natural frequencies of a lumped-mass system will always be lower than those of the represented system, with the degree of approximation dependent on the quantity of mass points and connecting members in the idealization.
2. The stiffness (and thus normal modes) of a structure as stable as a truss will usually be well represented by a pin-jointed truss. Secondary stresses will not be found directly, but may be estimated from displacements.
3. The stiffness of a shear panel may be represented by a lattice of pin-connected members (Ref. 14).
4. Flexible supports may be represented by inserting members with appropriate stiffness at points of support.
5. Reactions may be computed as member stresses by inserting very stiff members at points of support.
6. A beam of varying section properties may be approximated by several beams of constant section. Care should be taken that the problem does not become ill-conditioned by making the stiffnesses of the small beam segments very large in comparison to other elements of the structure.
7. Members normal to the plane of a grid may be included by adding appropriate stiffnesses to the matrix of the grid in the normal direction.
8. Pin-jointed members in a rigid-jointed frame may be represented with zero moment of inertia. A joint at which there is no moment resistance must be restrained against rotation for stability.
9. Loads applied at the interior of a bending member may be approximated by shear and fixed-end moments at its ends.
10. The validity of an idealization may be checked by comparison with a continuous structure idealization, or by model testing.
11. Increments of elements of the stiffness matrix may be input to the program; thus, the stiffness of structural components which are not conveniently idealized by a standard member may be included in an analysis.

### III. PROGRAMMING

#### A. Input Format

Input to the program is provided in the following blocks. An example of the input format is given in the sample problem, Appendix C.

##### 1. Control

##### 2. Joint coordinates

##### 3. Member properties

##### 4. Restraints

##### 5. Stiffness matrix elements (optional)

##### 6. Weights (optional)

##### 7. Loadings (optional)

##### 8. Accelerations (optional)

#### 0 Compute no normal modes

- 1 Compute lowest six mode shapes, normalized to input accelerations, and dynamic stresses
- 2 Compute lowest six mode shapes only, normalized to the largest component ( $N_{max} = 1$ ).

#### N<sub>o</sub> Output code

#### 0 No output of K, L, M matrices

#### 1 Output K, L, M matrices

#### N<sub>o</sub> Quantity of stiffness matrix elements to be altered

#### N<sub>o</sub> Modulus code

#### 0 Elastic properties for all members given by next two words

#### E Elastic properties of each member input separately

#### P Poisson's ratio

#### S Joint Coordinates

#### Joint number (must be listed consecutively starting with 1)

#### N<sub>o</sub> Joint coordinates, in.

#### A<sub>1</sub> A<sub>2</sub> A<sub>3</sub> A<sub>4</sub> A<sub>5</sub> A<sub>6</sub>

#### Z<sub>1</sub> Z<sub>2</sub> Z<sub>3</sub> Z<sub>4</sub>

#### N<sub>o</sub> Properties

The most convenient sheet on which to enter data for specifying coordinate sizes or active columns. Each row is punched on one card, with a maximum of nine words per card. Where word is not required, or input for a problem has, as many as ten blank spaces are always read as 0. Most of the input is written with four-digit numbers (integers) in the first two columns and floating-point numbers (rational numbers) in succeeding columns. When a number can have a fractional part, it must be written with a decimal point, regardless of whether the fractional part is present.

#### 1. General

#### N<sub>o</sub> Problem number (six digits or less)

#### N<sub>o</sub> Quantity of joints in structure

#### N<sub>o</sub> Quantity of members in structure

#### N<sub>o</sub> Quantity of loadings

#### N<sub>o</sub> Weight code

#### 0 No weight input

#### 1 Weight input included

#### N<sub>o</sub> Quantity of joints having one or more components of restraint

#### N<sub>o</sub> Degree of freedom per joint

#### N<sub>o</sub> Normal-mode code

Bending and torsional rigidity of rectangular slab strip will be computed from W and H if  $A_1 = I = 0$  is input. The effective moment of inertia, including stiffening due to adjacent strips, is

$$I_{eff} = \frac{W^3 P}{12(1 - \mu^2)}$$

and

$$K = \frac{W^2 P}{3}$$

#### 4. Materials

$\epsilon_{1x}$	Components of unit vector $\hat{\epsilon}_x$ , defining direction of principal axis of cross section
$\epsilon_{1y}$	
$\epsilon_{1z}$	
$\epsilon_{2x}$	
$\epsilon_{2y}$	
$\epsilon_{2z}$	
$\epsilon_{3x}$	
$\epsilon_{3y}$	
$\epsilon_{3z}$	

#### 5. Restraints

1	Joint member (may be listed in any order)
2	Restraint code (integer)
3	No restraint
4	1st component of deflection at joint / is 0. The order of deflection components at a joint in each structure type is as follows:
5	a. Planar member, three-dimensional, rigid-jointed, circular cross-section members
6	b. Planar member, three-dimensional
7	$\epsilon_{11}$ = displacement in $\hat{x}_1$ direction
8	$\epsilon_{12}$ = displacement in $\hat{x}_2$ direction
9	$\epsilon_{13}$ = displacement in $\hat{x}_3$ direction
10	$\epsilon_{21}$ = rotation about $\hat{x}_1$ axis
11	$\epsilon_{22}$ = rotation about $\hat{x}_2$ axis
12	$\epsilon_{23}$ = rotation about $\hat{x}_3$ axis
13	c. Planar member, rigid joints, loaded in-plane dimensions
14	$\epsilon_{11}$ = displacement in $\hat{x}_1$ direction
15	$\epsilon_{12}$ = displacement in $\hat{x}_2$ direction
16	$\epsilon_{13}$ = displacement in $\hat{x}_3$ direction
17	$\epsilon_{21}$ = rotation about $\hat{x}_1$ axis
18	$\epsilon_{22}$ = rotation about $\hat{x}_2$ axis
19	d. Planar member, right joints, loaded normal-to-plane (grid)
20	$\epsilon_{11}$ = displacement in $\hat{x}_1$ direction

#### 6. Three-dimensional, rigid-jointed members

#### A<sub>1</sub> = D, outside diameter

#### A<sub>1</sub> = T, wall thickness

#### A<sub>1</sub> = A, section area

#### A<sub>1</sub> = I, section moment of inertia in any direction

If  $A_1 = 0$ , then the area and moment of inertia will be given by  $A_0$  and  $I_0$ , not calculated from D and T.

#### c. Two-dimensional, rigid-jointed members

#### A<sub>1</sub> = A, section area

#### A<sub>2</sub> = I, section moment of inertia

#### A<sub>3</sub> = D, outside diameter

#### A<sub>4</sub> = T, wall thickness

#### A<sub>5</sub> = A, section area

#### A<sub>6</sub> = I, section moment of inertia

Area and moment of inertia of circular section will be computed from D and T and if  $A_1 = A = 0$  is input.

#### d. Two-dimensional grid

#### A<sub>1</sub> = I, section effective moment of inertia

#### A<sub>2</sub> = K, section torsional stiffness per unit length (Ref. 11)

#### A<sub>3</sub> = W, width of rectangular section

#### A<sub>4</sub> = H, depth of rectangular section

$\epsilon_{1x}$	Components of unit vector $\hat{\epsilon}_x$ , defining direction of principal axis of cross section
$\epsilon_{1y}$	
$\epsilon_{1z}$	
$\epsilon_{2x}$	
$\epsilon_{2y}$	
$\epsilon_{2z}$	
$\epsilon_{3x}$	
$\epsilon_{3y}$	
$\epsilon_{3z}$	

Properties are normally entered on one line (one card per member) except when modulus code,  $N_{11} = 1$ : in that case, a card containing  $\epsilon$  and  $r$  must be included in the deck for each member.

#### 3. Member Properties

#### Properties are normally entered on one line (one card per member) except when modulus code, $N_{11} = 1$ : in that case, a card containing $\epsilon$ and $r$ must be included in the deck for each member.

#### 4. Member Cards

Properties are normally entered on one line (one card per member) except when modulus code,  $N_{11} = 1$ : in that case, a card containing  $\epsilon$  and  $r$  must be included in the deck for each member.

$\alpha_x$  = rotation about  $x$  axis  
 $\alpha_y$  = rotation about  $y$  axis

### 3. Stiffness Matrix Elements

To account for the effect of structural elements which cannot be simulated by members of the 'type' with which an analysis is being performed, increments to elements of the matrix may be input. This block may be input only if the control parameter  $N_m \neq 0$ .

$\begin{bmatrix} 1 & 1 & 1 & 1 & \Delta k_{ij} \end{bmatrix}$

Now and column, respectively, of revised element in contracted stiffness matrix (inserted after rows and columns have been deleted to account for constraints).

Incremental change to element  $k_{ij}$  of original stiffness matrix. The new element,  $k_{ij} = k_{ij} + \Delta k_{ij}$ .

### 6. Weights

$\begin{bmatrix} 1 & \text{Blank} & |W_1| & |W_2| & |W_3| & |W_4| & |W_5| & |W_6| \end{bmatrix}$

Joint number (may be listed in any order)

$W_i$ , i-th component of inertia at joint  $i$ . The order of translational inertia (lb) and rotary inertia (lb-in.) components is as specified for deflections in A4.

If normal modes are to be computed, finite (non-zero) inertia components should be specified for all degrees of freedom of the structure; the effect of a zero inertia is to produce accumulator overflow. (This is a peculiarity of the numerical procedure.) This block of input may be written only if the weight code,  $N_w = 1$ . If no headings follow ( $N_w = 0$ ), the last card of weights must be followed by a card with 0 as its first word.

### 7. Loadings

Each loading is initiated by a card with 0 as the first word, and the final loading must be followed by a card with 0 as the first word.

The initial card has the format:

$\begin{bmatrix} 0 & \text{Blank} & |e_1| & |e_2| & |e_3| \end{bmatrix}$

Subsequent cards (which need not be given) have the format:

$\begin{bmatrix} 1 & \text{Blank} & |f_1| & |f_2| & |f_3| & |f_4| & |f_5| & |f_6| \end{bmatrix}$

- 6. Component of translational acceleration load on the structure as a rigid body in the  $i$ th coordinate direction. (The sense of the load is opposite to the direction of acceleration.) The effect of specifying  $e_i$  ( $f_i$ ) is to multiply the item in component  $W_i$  at each joint by  $a_i$ . Note that this has no physical meaning for rotational components.
- 7. Component of concentrated load on joint  $i$  in the  $ik$  direction. Order of load components is as specified for deflections (see A4).
- 8. Accelerations

If the normal-mode code,  $N_m = 0$  or 2, this block must be omitted. If  $N_m = 1$ , deflections and dynamic stresses will be computed. In this case, six word must be given in the following format (one for each mode in in order):

$\begin{bmatrix} 1 & 1 & 1 & 1 & q_1 & q_2 \end{bmatrix}$

Joint number

Component direction number as specified for deflections (see A4d):

$q_1$  = Acceleration ( $\ddot{s}$ ) of joint  $i$  in direction 1. If  $j = 0$ , the acceleration  $q_1$  applies to the maximum deflection component in the mode shape. The mode shape is renormalized with the factor  $q_1 \cdot \sqrt{u_{11}/\omega_1^2}$  before output and stress calculation. Rotatory accelerations have no meaning in this application.

### Size Limitations

Degree of freedom of structure  $\leq 130$   
 Joints in structure (free or fixed)  $\leq 60$   
 Members in structure  $\leq 200$   
 Components of restraint  $\leq 100$   
 Loadings  $\leq 6$   
 Joints  $\times$  degree of freedom per joint  $\leq 180$

### B. Output Format

The output is printed in the following divisions. An example of the output format is given in the sample problem, Appendix C.

1. Stiffness matrix, weight (mass) matrix, load matrix printed columnwise, ten words per line.

### D. New Charts

Symbols used in the matrix, weight (mass) matrix, load matrix programs are not the same, but correspondence between the two is indicated where possible. Program symbols are

- 2. Static deflections. Each column corresponds to one loading; deflections at each joint follow the joint number in the order specified in A4.
- 3. Static number stresses. Formats for various member types are given in Appendix B.

- 4. Convergence data. The results of accuracy tests discussed in Section II-G are printed under appropriate headings.

- 5. Frequencies, computed from the eigenvalues of the matrix  $C$  (see Section II-F), assuming input of mass in pound units and dimensions in inch units:

$$f_n = \frac{1}{2\pi} \sqrt{\frac{E}{\rho_n}} = 5.128518 \sqrt{\frac{1}{\rho_n}}$$

- 6. Dynamic member stresses. Formats for various member types are given in Appendix B.

### C. Operator Instructions

The program comprises 13 FORTRAN-coded routines plus library routines. All routines are common to analyses of all four structure types except STRIX and STRES. Program and COMMON storage amounts to 30,000 words (decimal).

As compiled with the Jet Propulsion Laboratory system, there are no program stops. The input deck typically includes:

1. System instructions
2. Machine language program
3. DATA card
4. Data
5. End of file

The following type assignments are made:

FORTRAN	Logical	User	NORM
5	A3	BCD Input	ORTH
6	A3	BCD output	
14	BS	Binary temporary storage	

FORTRAN	Logical	User	NORM
STM			71
STRAX			
STRES			

used in the flow charts presented in this section. Routines and subroutines are listed in the master flow chart by number.

FORTRAN operational symbols used in the diagrams indicate:

1. Replacement  $A = B$
2. Subscript  $A(I)$
3. Arithmetic  $A+B$
4. Exponentiation  $A^B$

The following simplifying notation is used where applicable:

1. Subroutine names and arguments are underlined.
2. Matrix columns are indicated by a single subscript, e.g.,  $V(1)$  is the 1st column of the array  $V$ .
3. An array  $A$  is cleared by the notation  $A = 0$ .

The flow chart diagram the following routines:

1. Purpose
2. Name Length
- (Main) 1102 Generate matrices and control solution.
- DEPOT 205 Expand and output deflection matrix.
- DEFL 230 Compute  $U = K F$ .
- EIG 736 Control solution of  $KU = \omega^2 MU$ .
- MUL 81 Multiply vector times matrix.
- RICH 68 Control one vector iteration.
- INV 208 Invert symmetric matrix.
- KLR 47 Compute maximum change in vector components between two iterations.
- ORTH 40 Normalize vector to largest component.
- ORT 71 Orthogonalize one vector to another, compute Rayleigh's Quotient.
- QBLK 60 Compute  $M^{-1} K M^{-1} = A$ .
- Varie 208 Generate stiff-mass matrix for a structural member.
- Varie 208 Compute and output member stresses.

JPL TECHNICAL MEMORANDUM NO. 33-78

Common storage contains the following arrays:

Name	Dimension	Used For	JTA	$200 \times 1$	Member numbers
			JTB	$200 \times 1$	
AK	$130 \times 130$	Storage of stiffness matrix, static deflection vector, and mode shapes	A	$200 \times 9$	Member properties
			X	$60 \times 3$	Joint coordinates
AL	$180 \times 6$	(1) Generation of stiffness matrix (2) Load vectors (3) Expanded deflection vectors and eigenvectors (4) Eigenvectors during iteration	LDEL	$100 \times 1$	List of rows (columns) to be deleted (restraint components)
( = SM = VC)*	$(135 \times 8)^*$		F	$12 \times 1$	(1) Input buffer for weights and loads (2) Eigenvalues and frequencies
AM	$180 \times 1$	Mass (weight) vector	V	$3 \times 130$	Eigenvectors during iteration

\*Arrows may have alternate names and dimensions in different routines.

STIFFNESS MATRIX PROGRAM  
MASTER FLOW CHART

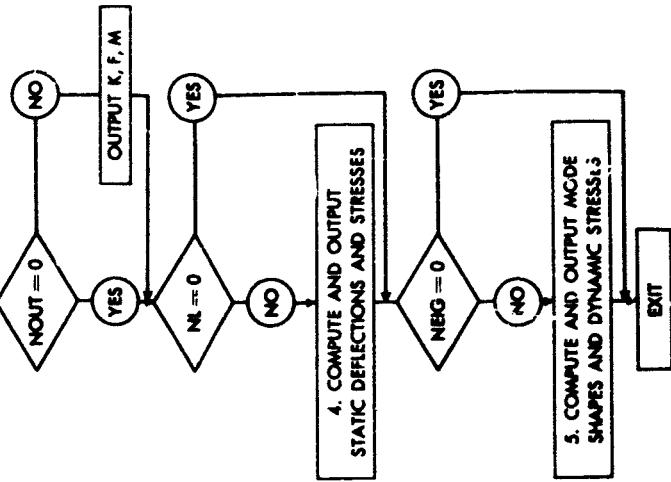
1. READ INPUT (CONTROL,  
GEOMETRY, RESTRAINTS)

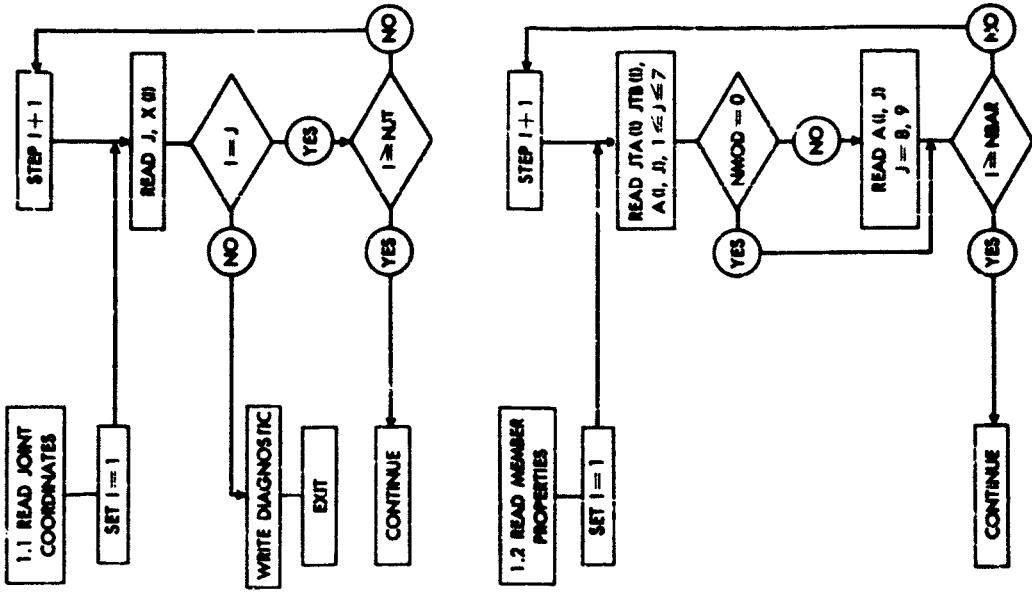
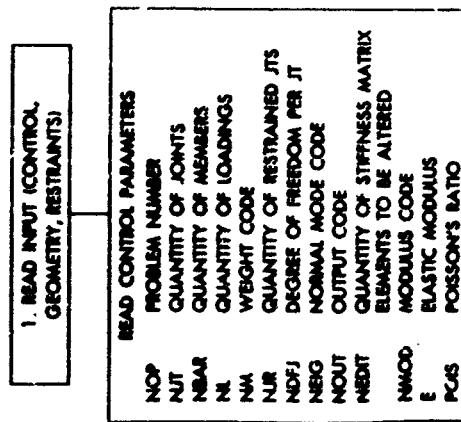
2. GENERATE STIFFNESS  
MATRIX (K)

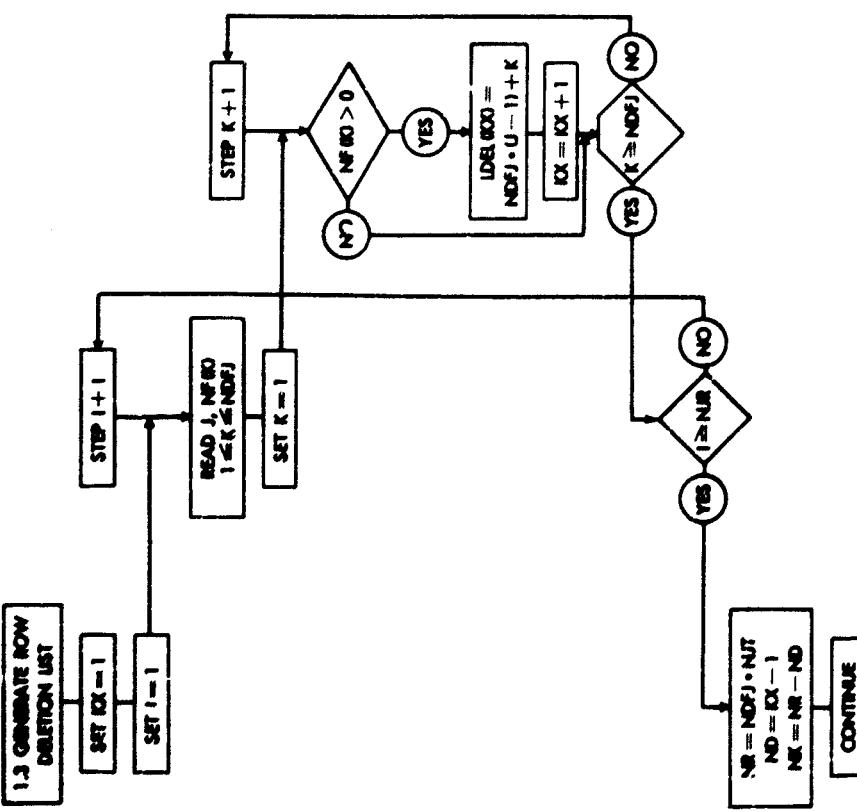
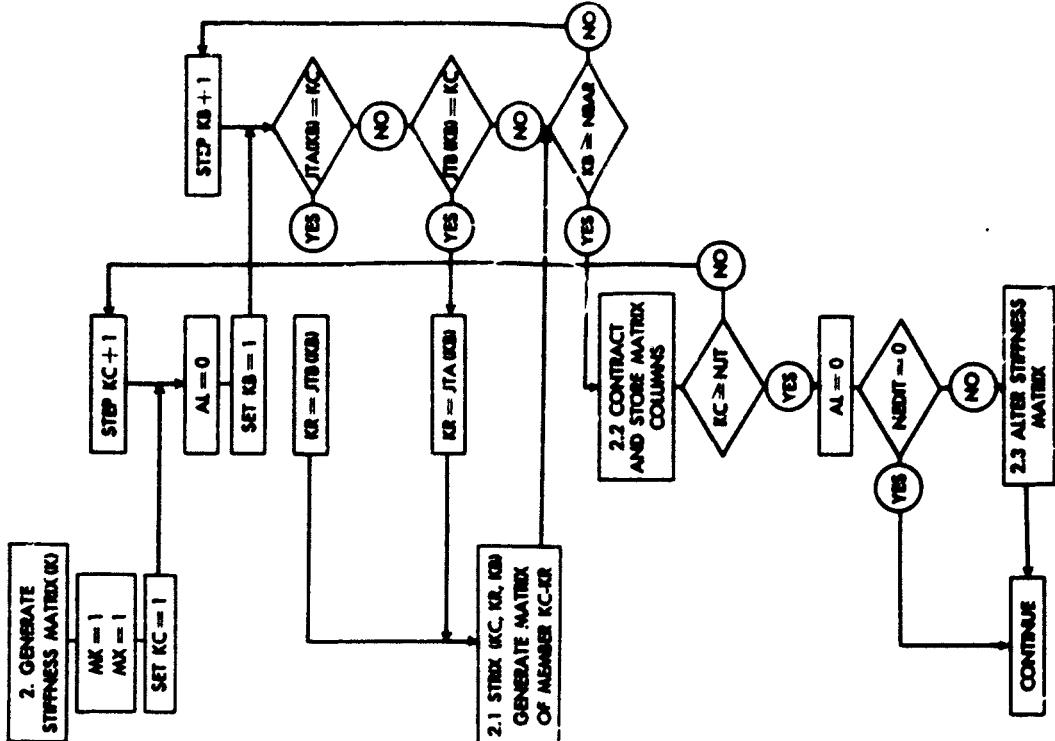
3. GENERATE LOAD AND  
INERTIA MATRICES (F, M)

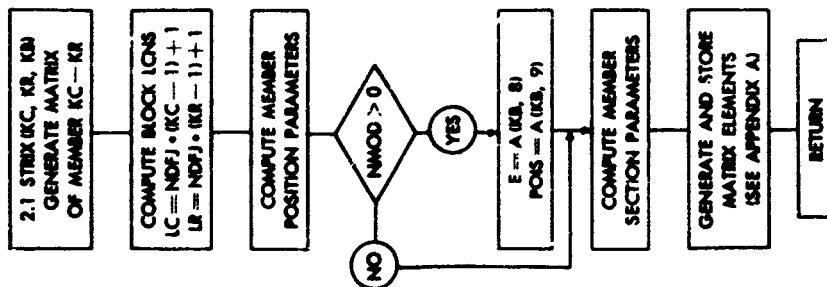
4. COMPUTE AND OUTPUT  
STATIC DEFLECTIONS AND STRESSES

5. COMPUTE AND OUTPUT MODE  
SHAPES AND DYNAMIC STRESSES

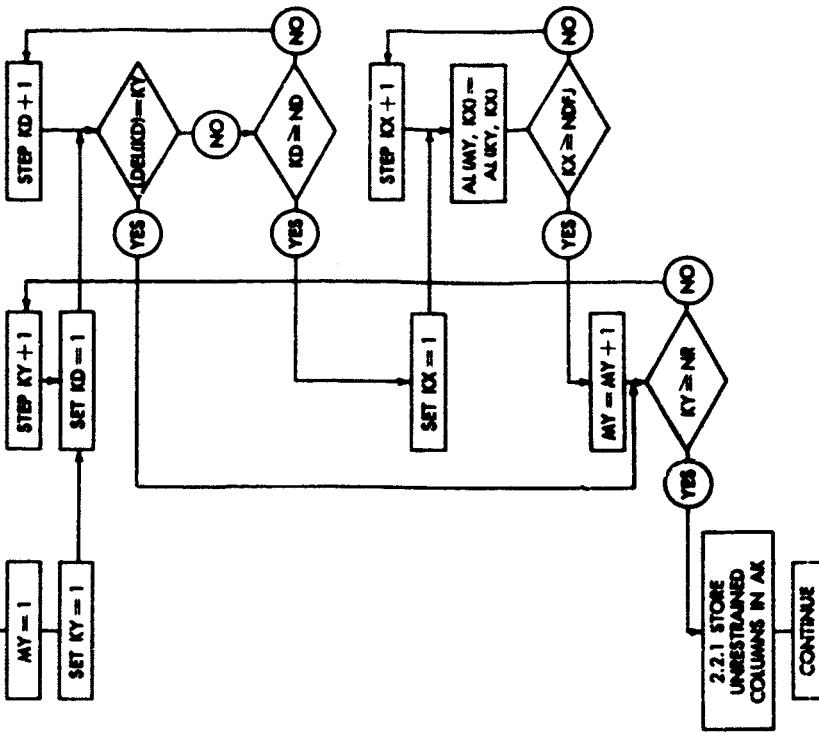


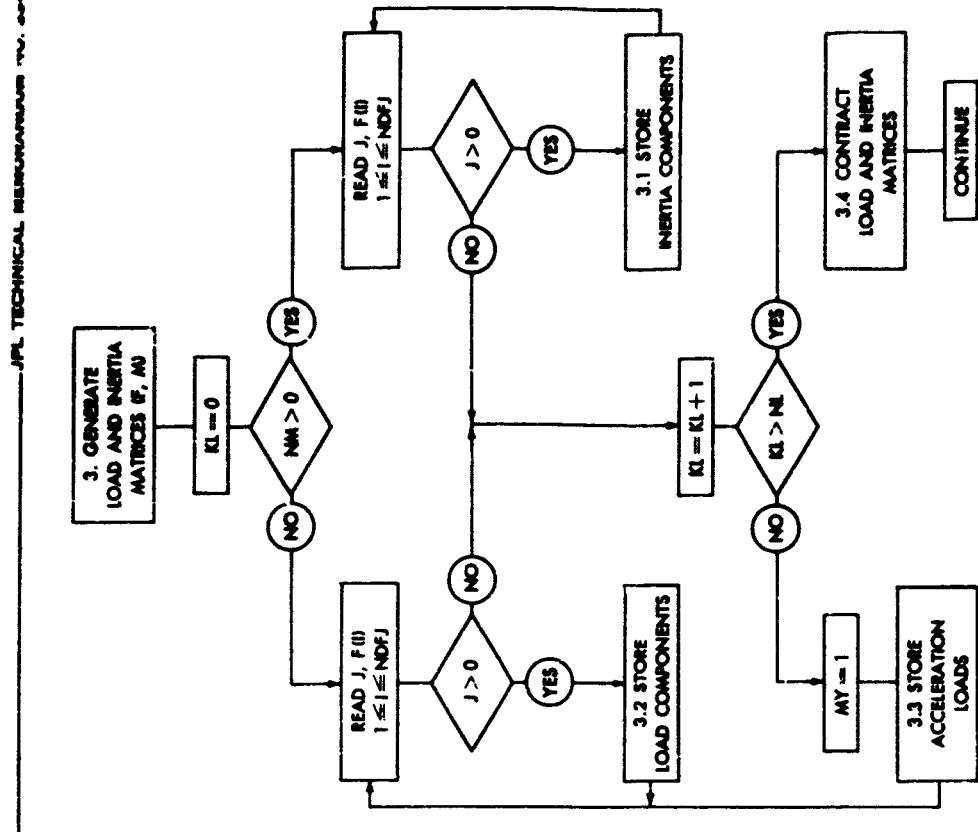
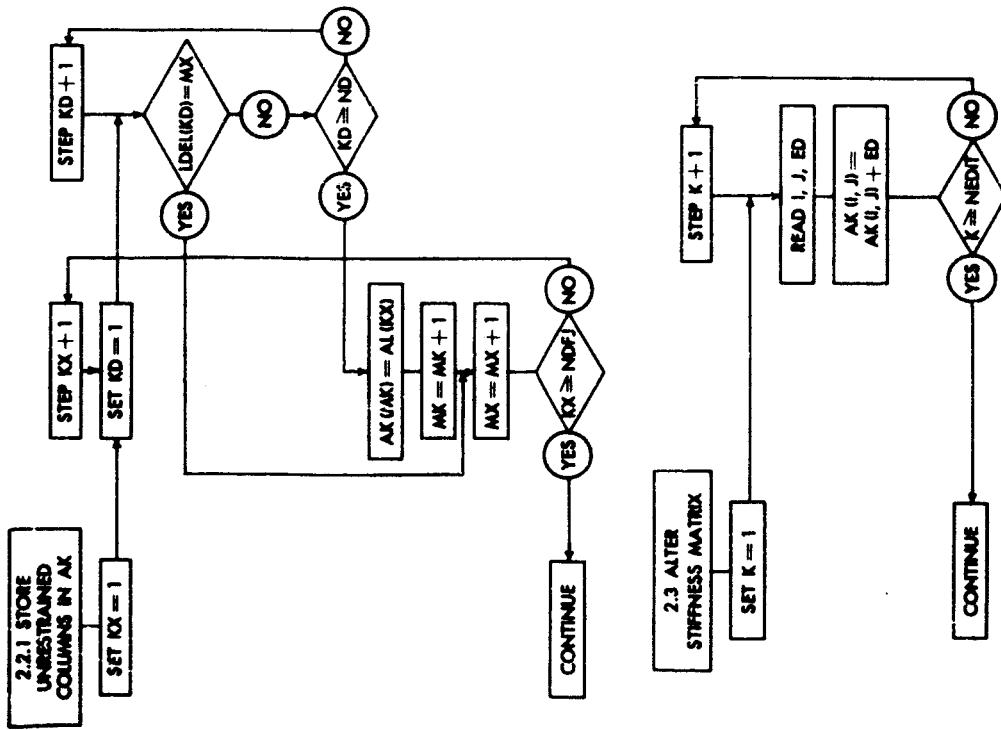


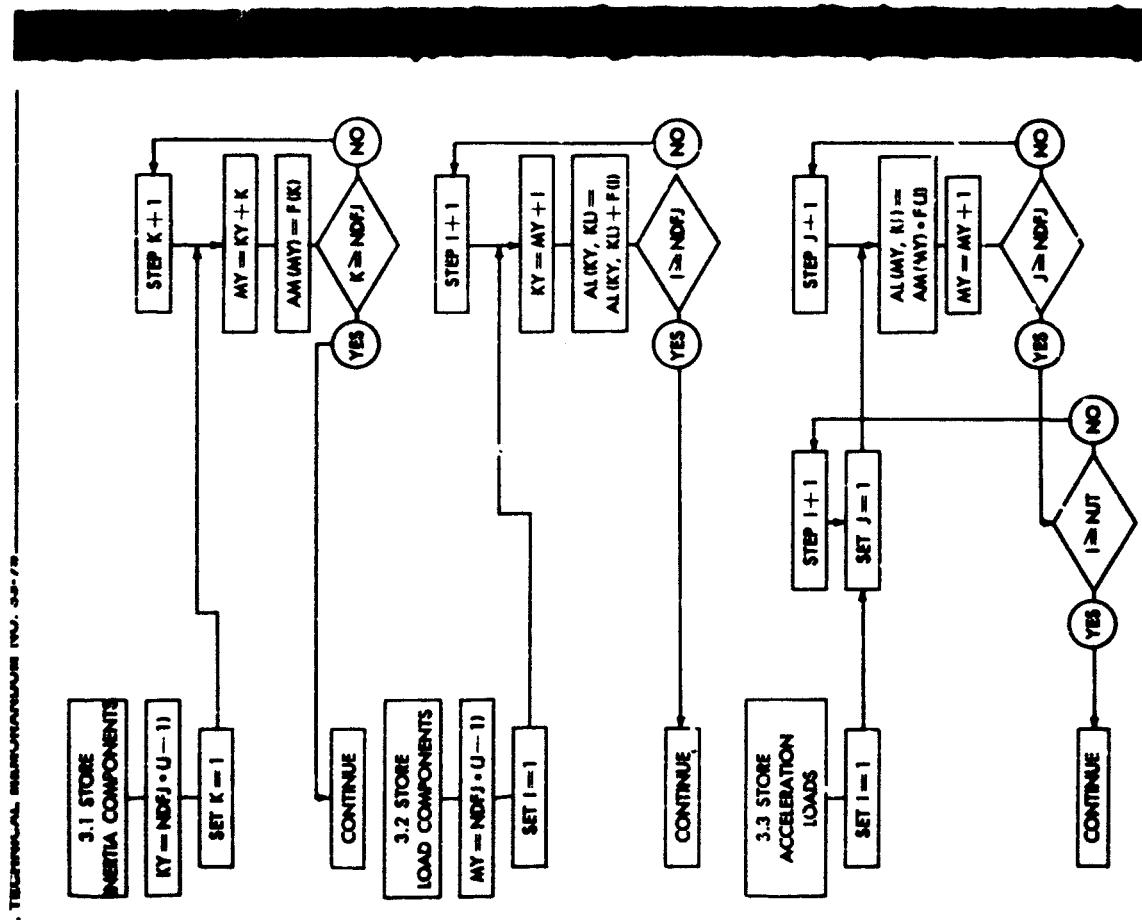
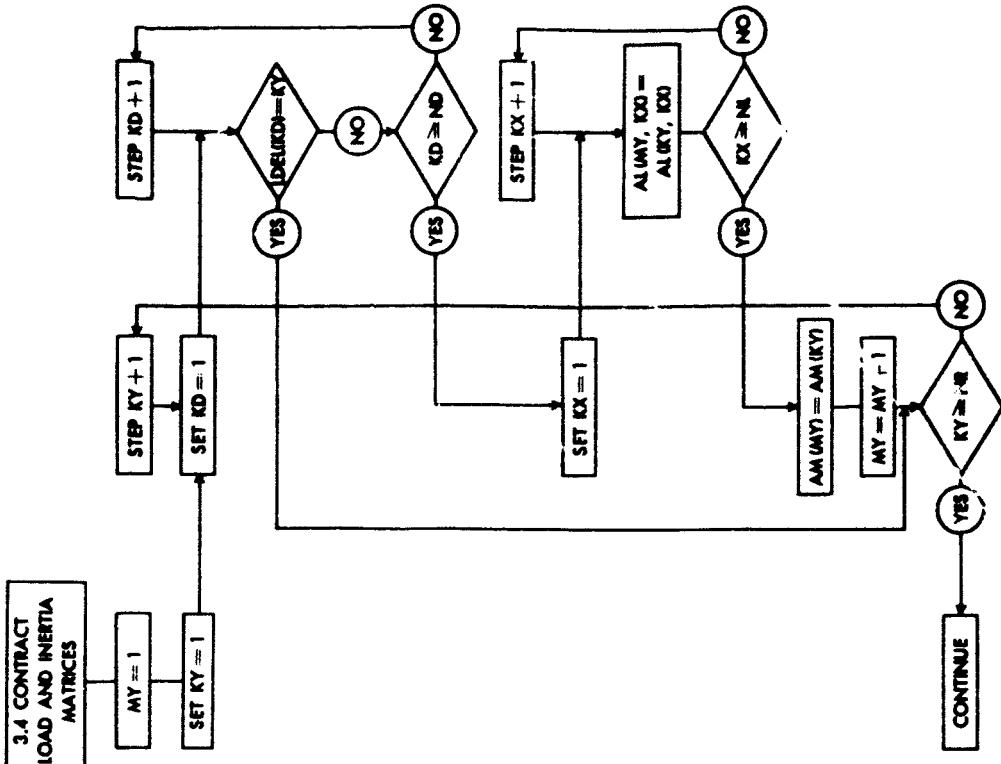


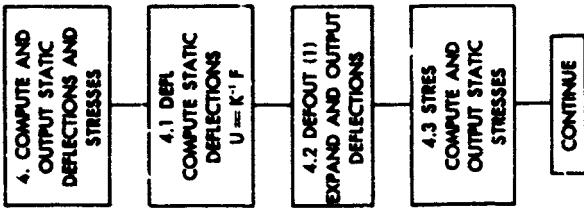


2.2 CONTRACT  
AND STORE  
MATRIX COLUMNS






**3.4 CONTRACT LOAD AND INERTIA MATRICES**




**4.1 DFL**  
COMPUTE STATIC DEFLECTIONS  
 $U = K' F$

$N1 = NK + 1$   
 $I1 = 1$   
 $NN = NK + N4$

SET  $J = 1$

STEP  $J + 1$   
 $\Delta K U + \Delta K0 = \Delta U$

IF  $J \geq N$  THEN

STEP  $J + 1$   
 $J1 = I + 1$   
SET  $J = J1$

STEP  $J + 1$   
 $\Delta K(0, JJ) =$   
 $\Delta K(0, JJ) / \Delta K(I, J)$

IF  $J \geq NN$  THEN

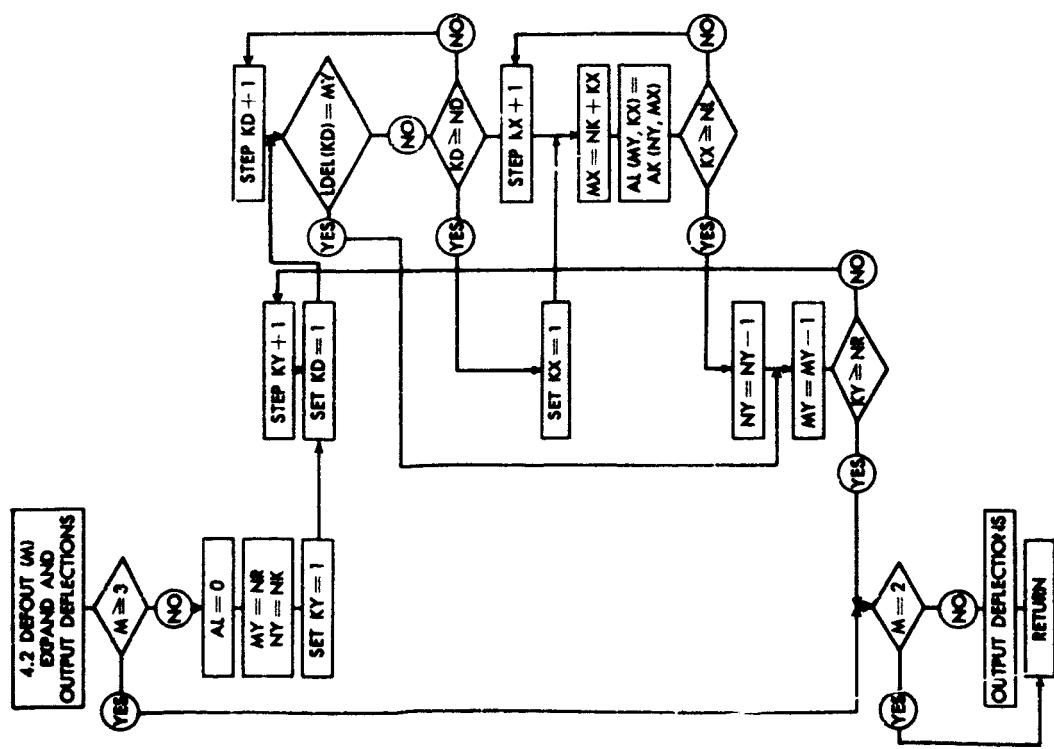
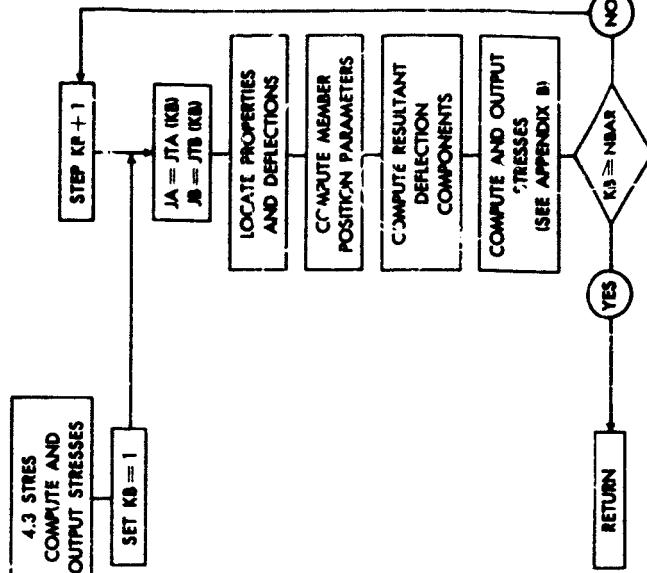
STEP  $K + 1$   
 $K = -1$   
IF  $K = -1$  THEN  
SET  $J = J1$   
IF  $\Delta K(0, JJ) = 0$  THEN  
RETURN

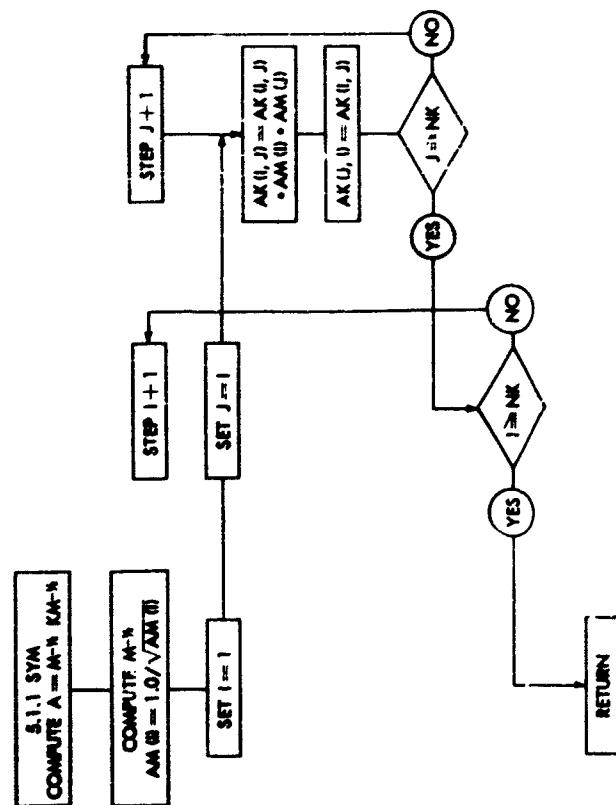
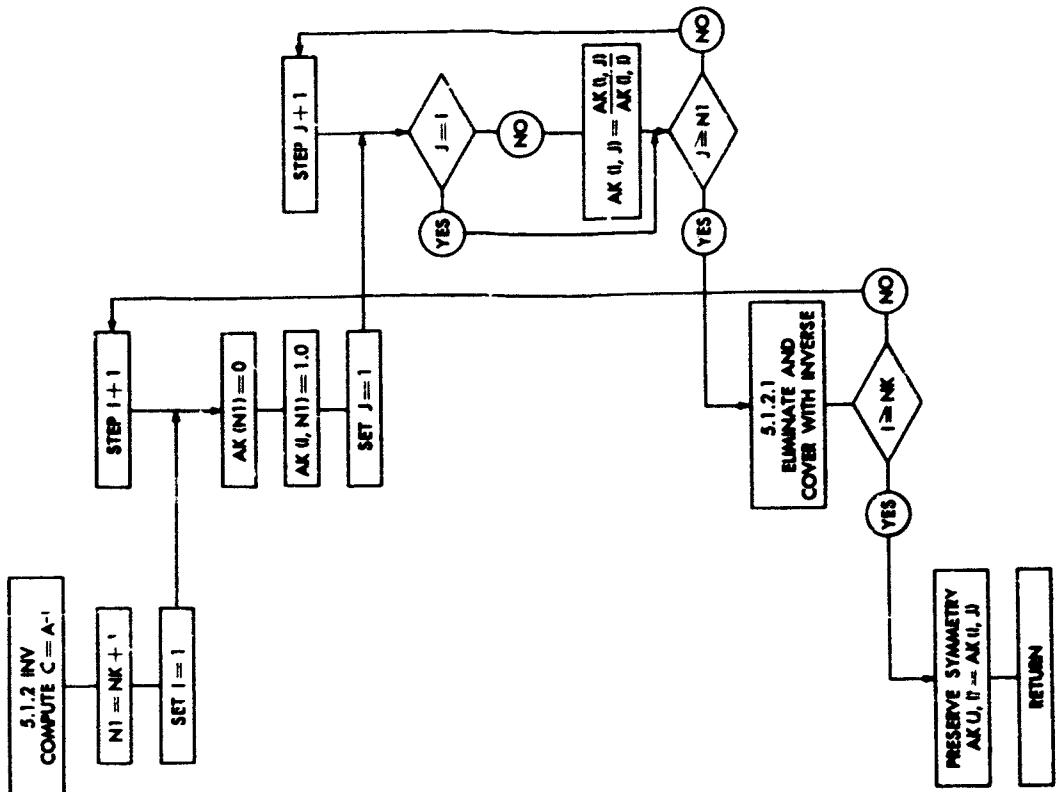
IF  $K \neq -1$  THEN  
SET  $J = J1$   
IF  $\Delta K(0, JJ) \neq 0$  THEN  
SET  $K = K + 1$   
IF  $K \geq N$  THEN  
RETURN

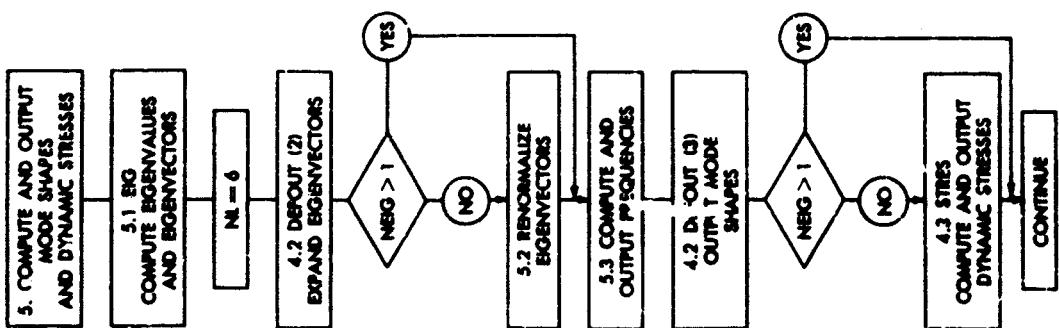
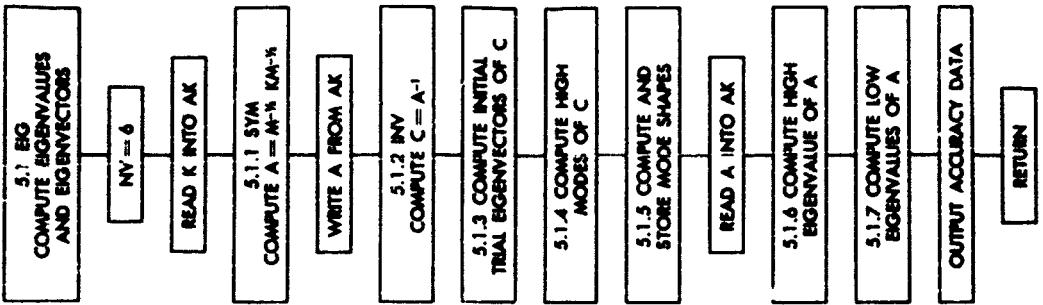
IF  $K \leq N$  THEN  
SET  $J = J1$   
IF  $J \geq NN$  THEN  
IF  $\Delta K(0, JJ) = 0$  THEN  
SET  $J = J1$   
IF  $\Delta K(0, JJ) \neq 0$  THEN  
SET  $K = K + 1$   
IF  $K \geq N$  THEN  
RETURN

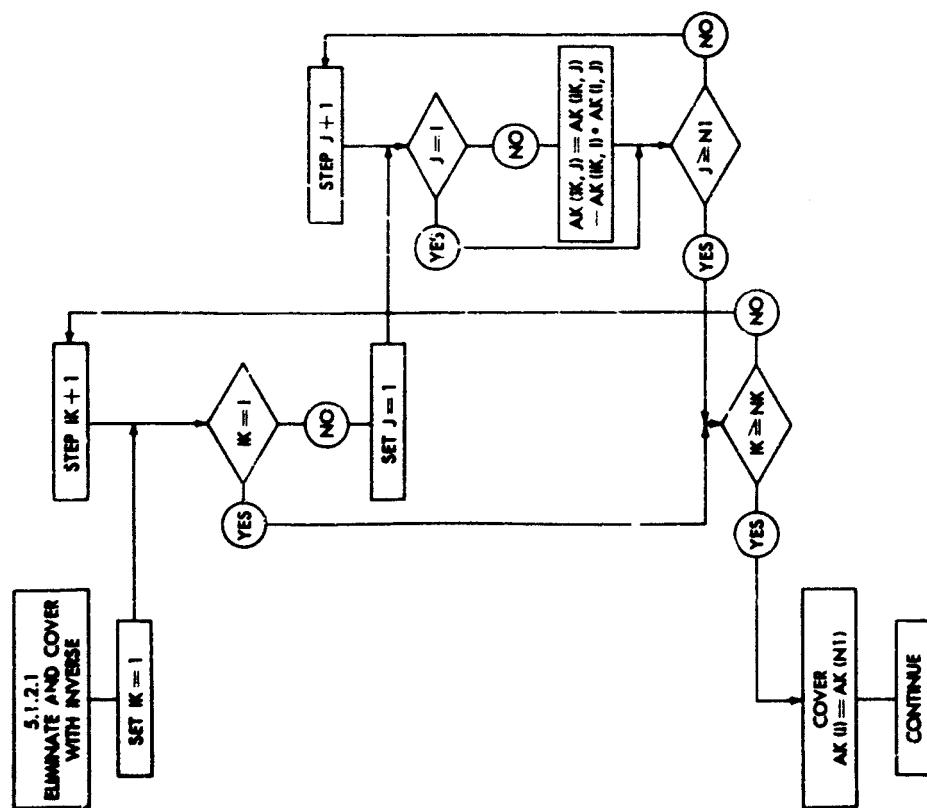
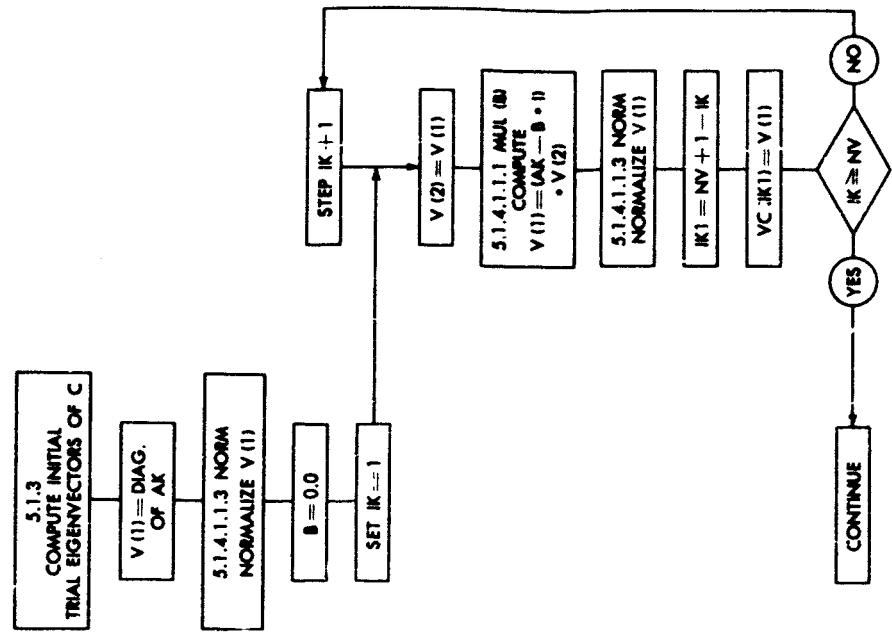
IF  $J \geq NN$  THEN  
SET  $J = J1$   
IF  $\Delta K(0, JJ) = 0$  THEN  
SET  $J = J1$   
IF  $\Delta K(0, JJ) \neq 0$  THEN  
SET  $K = K + 1$   
IF  $K \geq N$  THEN  
RETURN

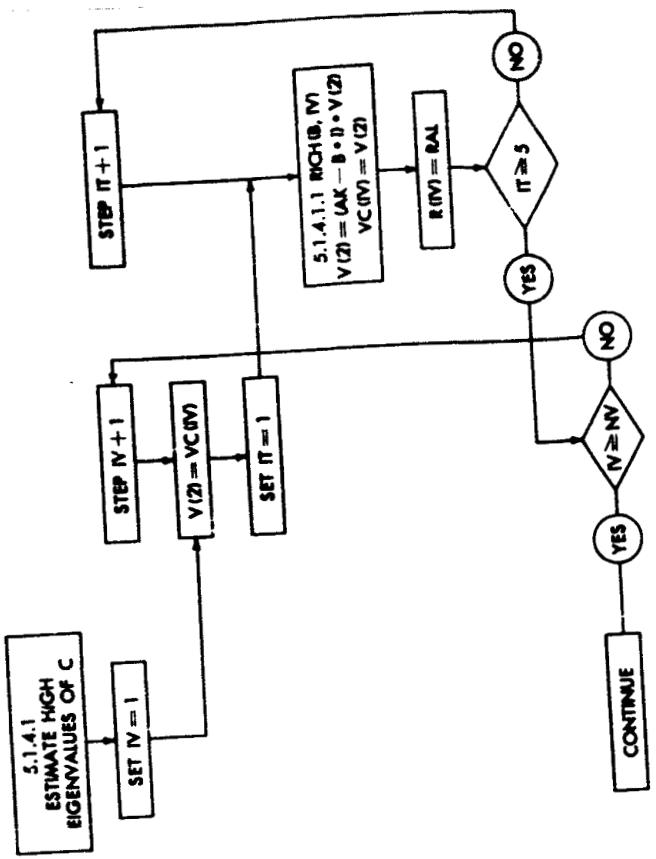
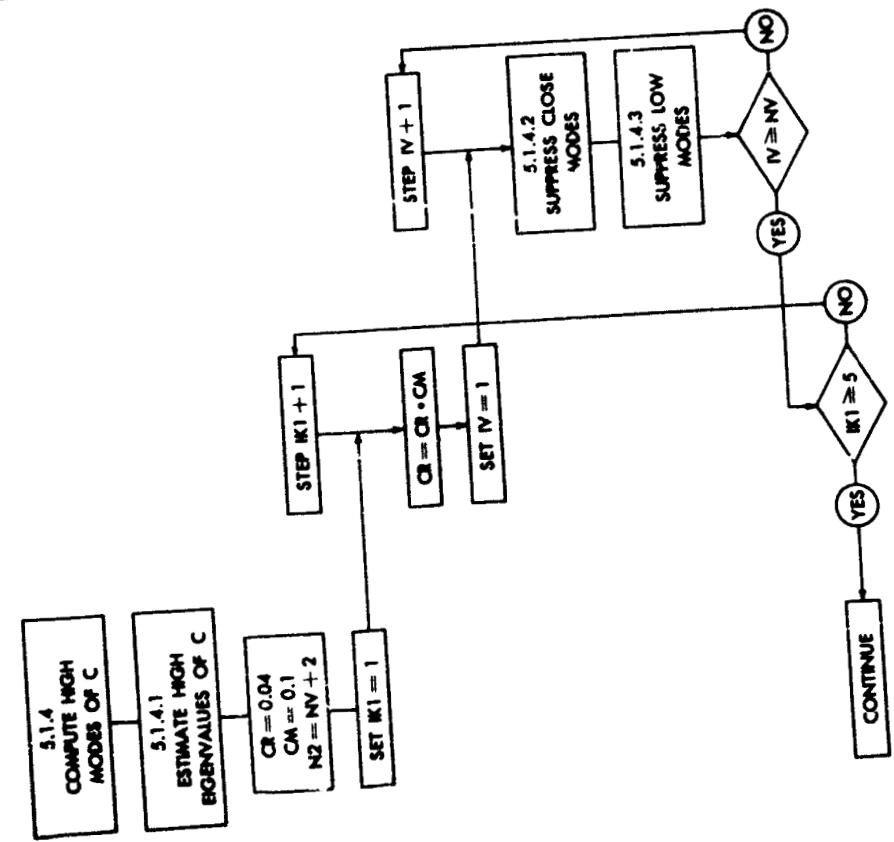
IF  $J \geq NN$  THEN  
SET  $J = J1$   
IF  $\Delta K(0, JJ) = 0$  THEN  
SET  $J = J1$   
IF  $\Delta K(0, JJ) \neq 0$  THEN  
SET  $K = K + 1$   
IF  $K \geq N$  THEN  
RETURN

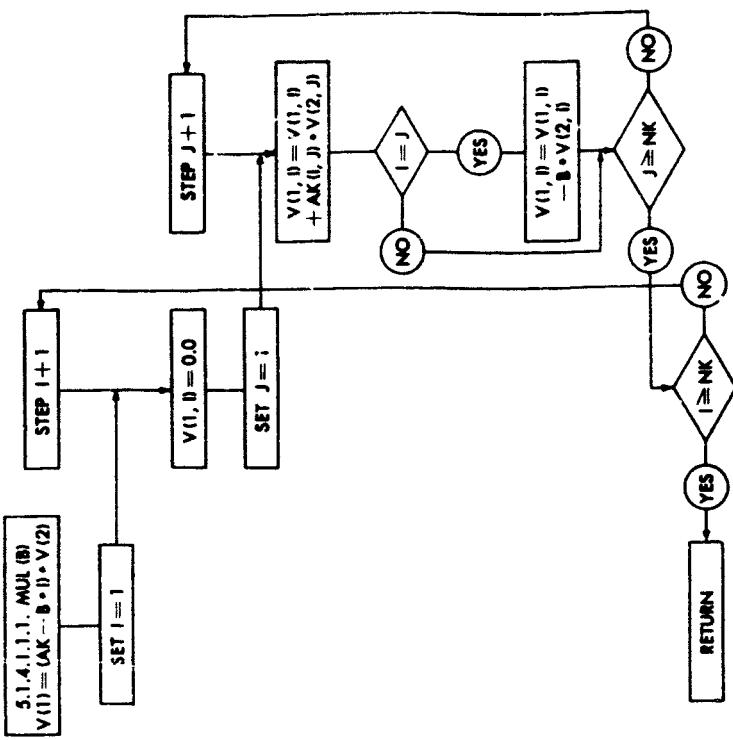
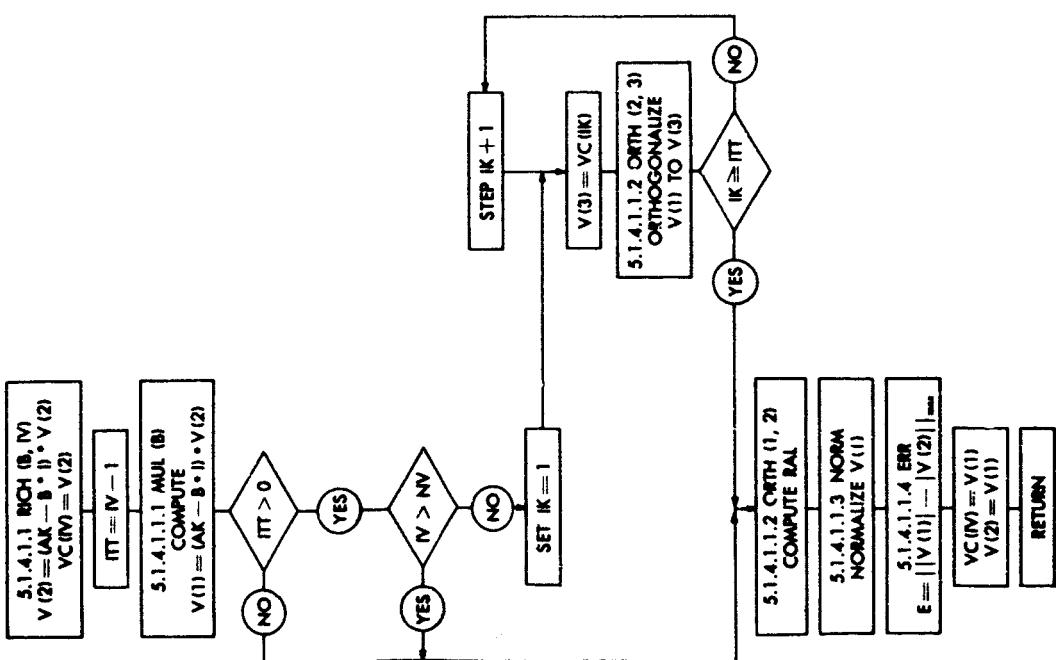
**4.3 STRES COMPUTE AND OUTPUT STRESSES**

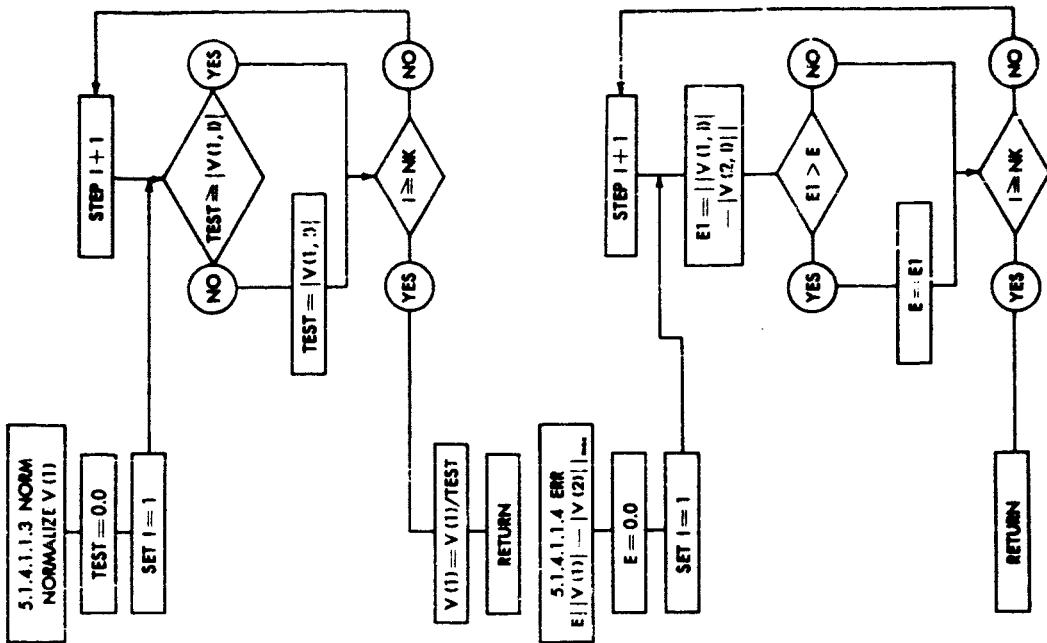
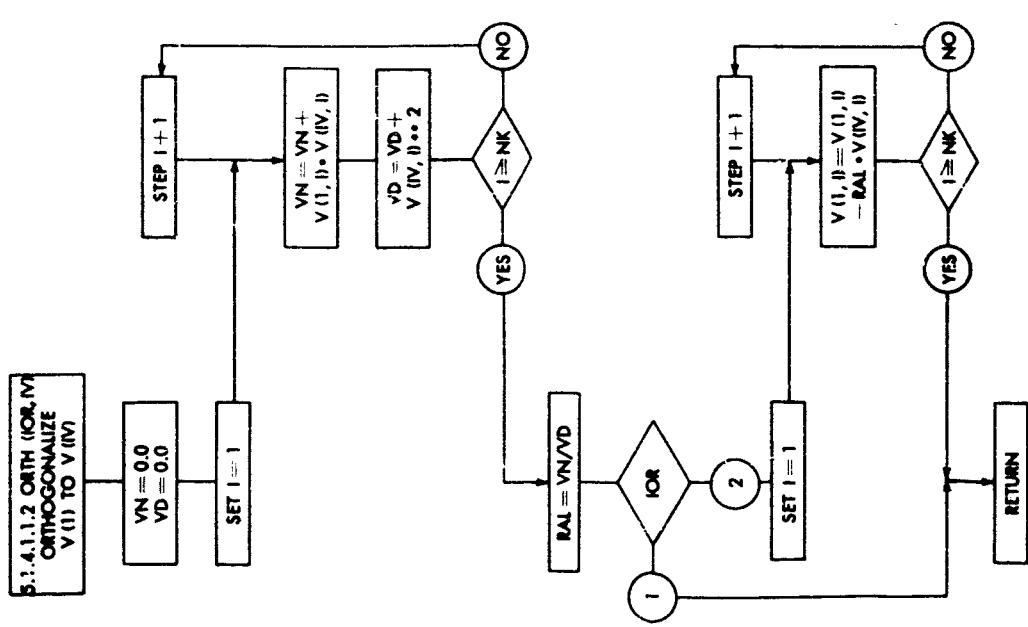


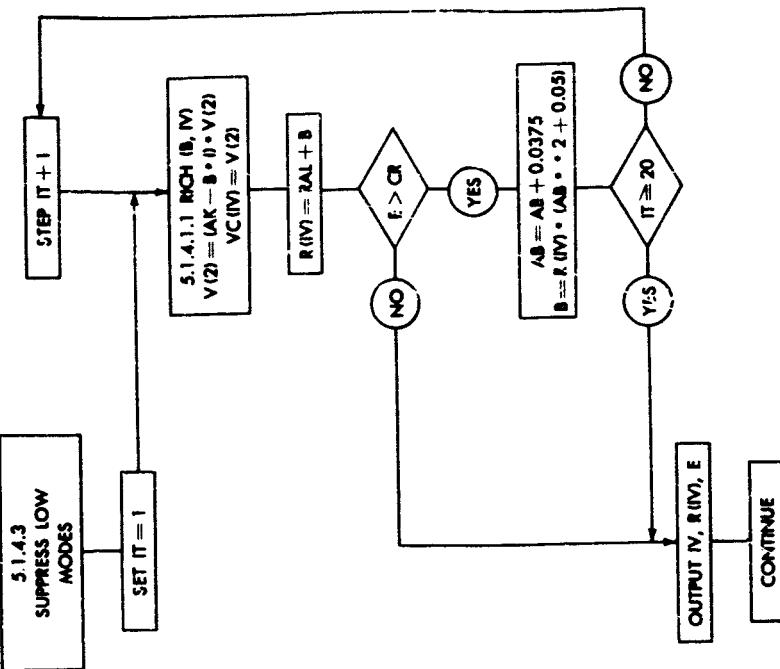
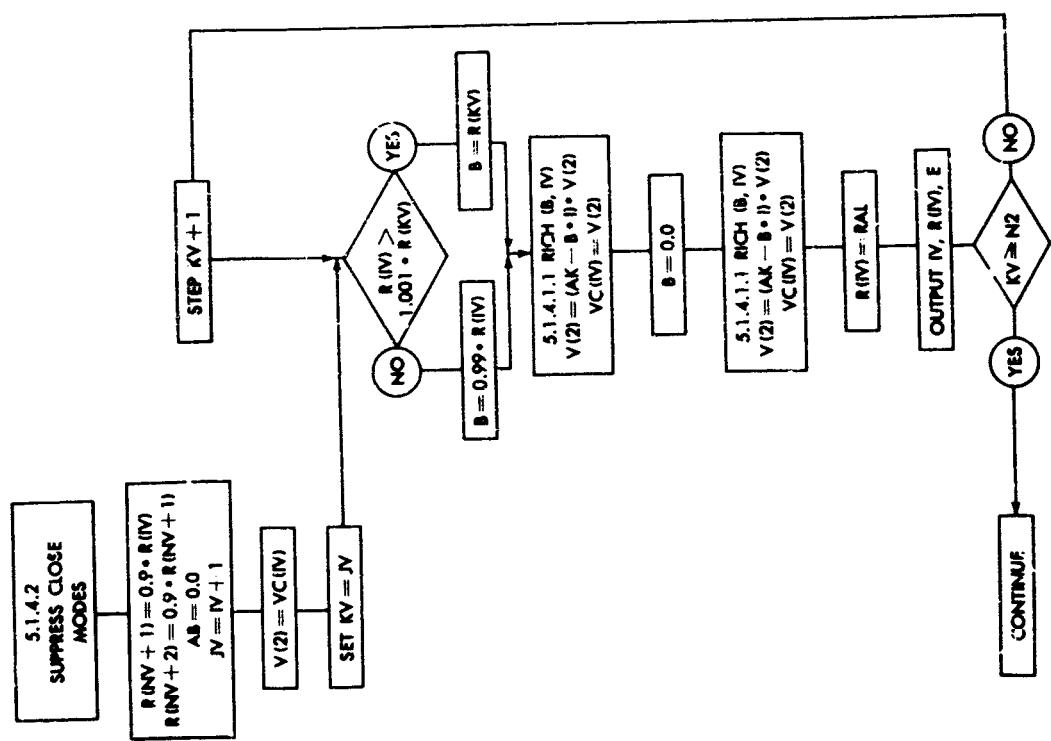


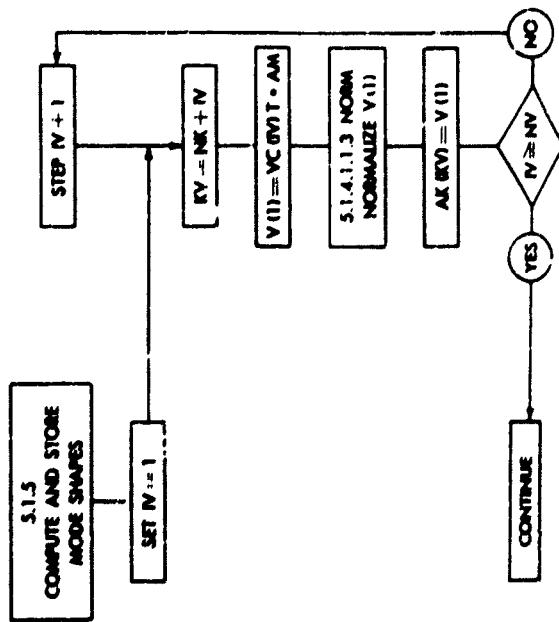
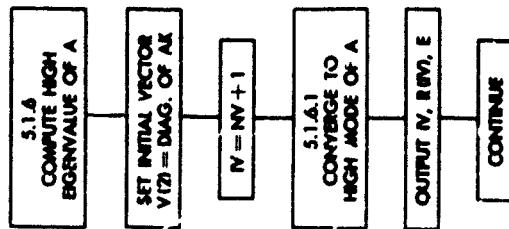


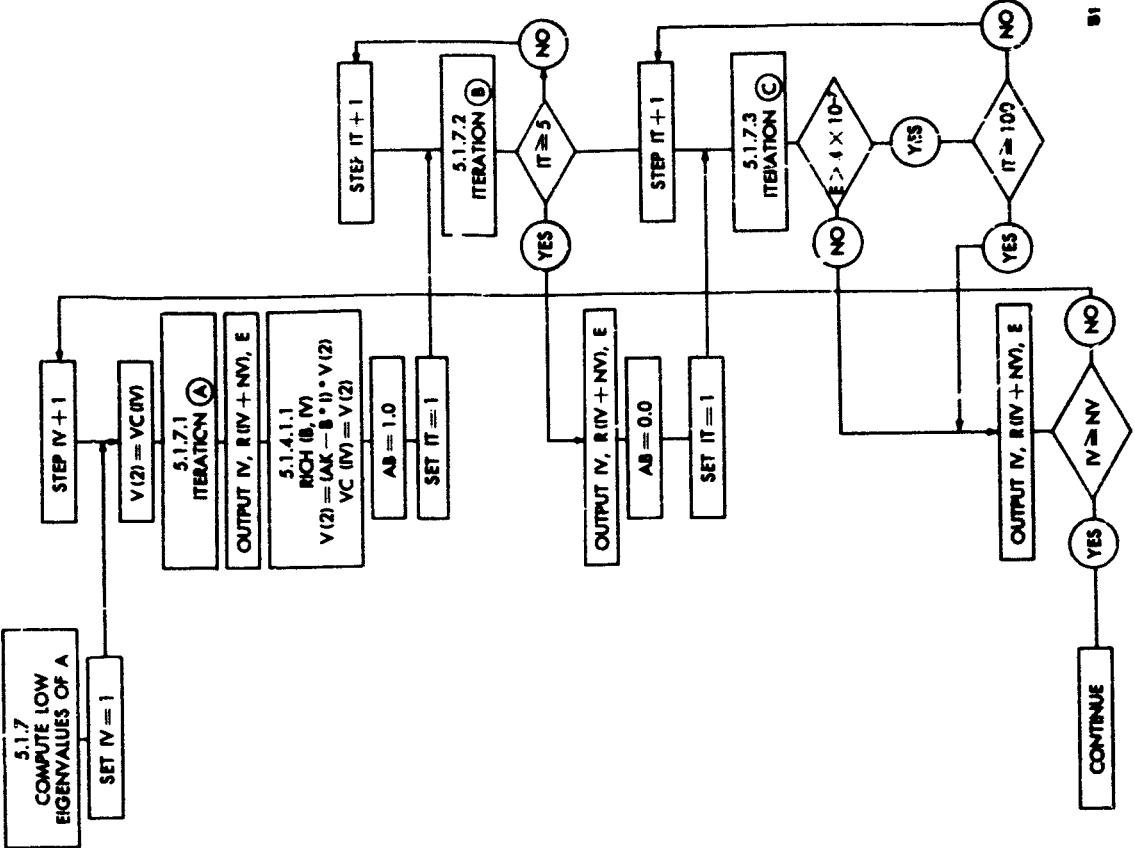
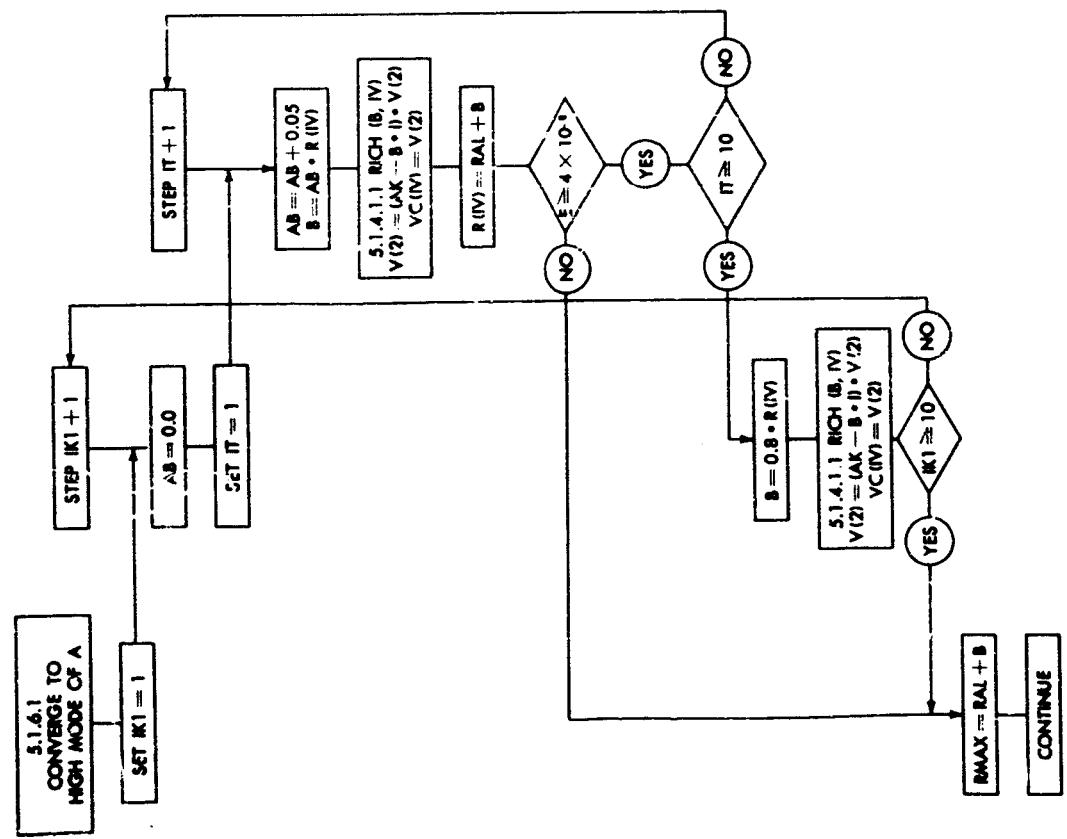


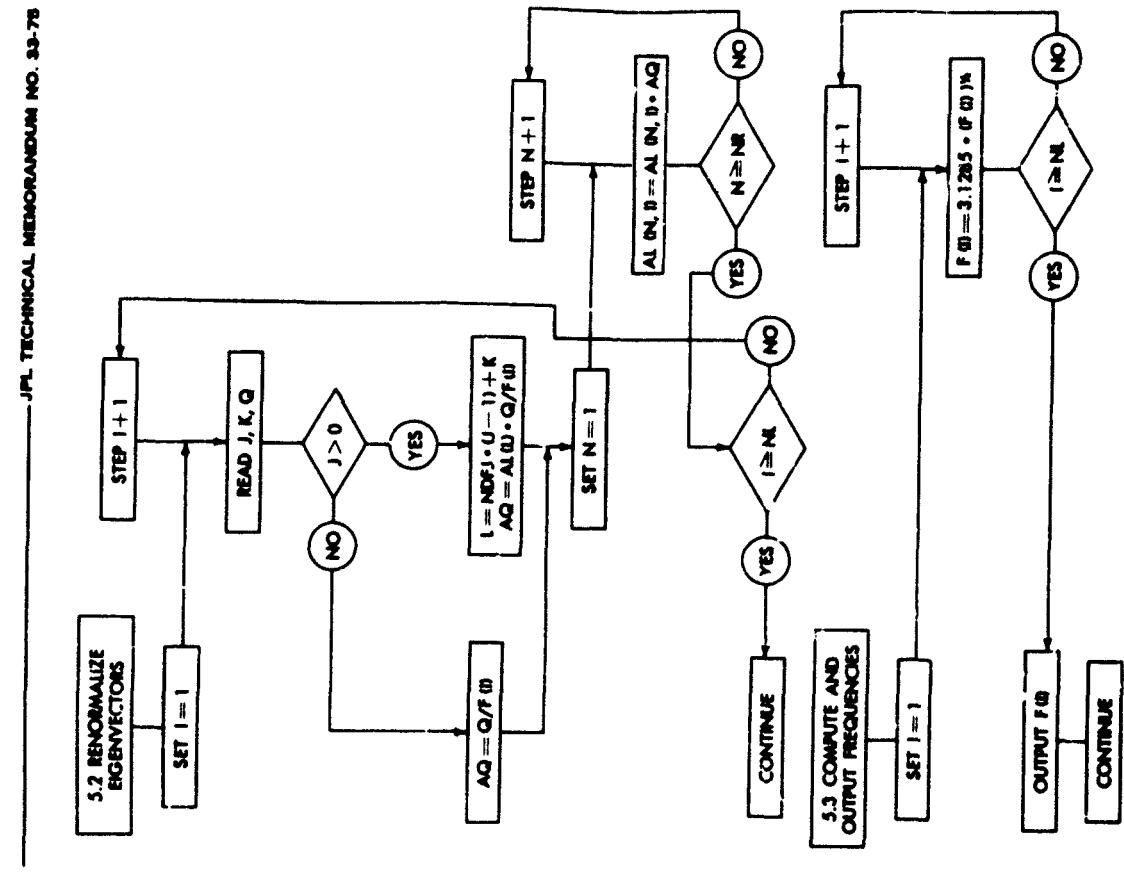
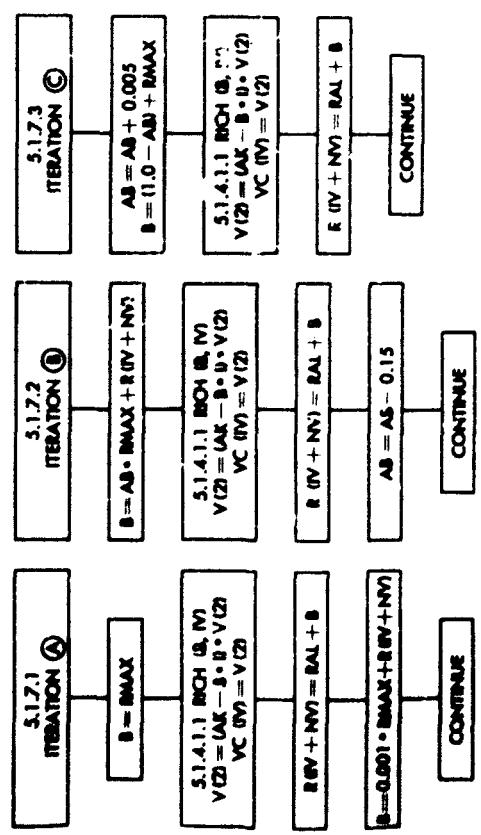












### E. Program Listing

A complete listing of the FORTRAN program required for the stiffness matrix structural analysis is presented in the following pages.

## E PROGRAM LISTING

## C STIFFNESS MATRIX ANALYSIS MASTER ROUTINE

```
COMMON/NL,NM,AK,AL,AM,JTA,JTD,X,NMDO,E,POLS,
```

```
      XNAT,SEARDOF,NDIM,NCODELIF
```

```
      DIMENSION AK(130,130),AL(130,6),AM(130,1),JTA(200),JTD(200),
```

```
      YAT(200,91),XY(90,91),D(130),PTZTMF(130),PTZTMF(130)
```

```
      L01 FORMAT(1B18.7E8.0)
```

```
      L02 FORMAT(1B18.7E8.0!)
```

```
      L03 FORMAT(1B18.0)
```

```
      L04 FORMAT(1I11.11,0!)
```

```
      L05 FORMAT(1I12H,INPUT,ERROR,READING,JOINT,COORDINATES)
```

```
      L06 FORMAT(1I12H,STIFFNESS,MATRIX,ANALYSIS,PROBLEM,16)
```

```
      L07 FORMAT(1I12H,LOAD,MATRIX)
```

```
      L08 FORMAT(1I12H,STATIC,DISPLACEMENTS)
```

```
      L09 FORMAT(1I12H,STATIC,MEMBER,STRESSES)
```

```
      L10 FORMAT(1I12H,MODE,SHAPEFACT)
```

```
      L11 FORMAT(1I12H,DYNAMIC, MEMBER,STRESSES)
```

```
      L12 FORMAT(1I12H,FREQUENCY,TEST)
```

```
      L13 FORMAT(1I16,2.5F12.2)
```

## C READ STRUCTURE INPUT, GENERATE STIFFNESS MATRIX

```
      READ INPUT TAPE $1,00,NOP,NJ,NBAR,NL,NM,NJK,NDE,J,NLG,NOUT
```

```
      WRITE OUTPUT TAPE $0,T0,T0P
```

```
      READ INPUT TAPE $1,103,NEDOF,NMDO,E,POLS
```

```
      NC=J1-J2, NJ=J2-J1
```

```
      READ INPUT TAPE $1,101,J,NDMP,(X11,K1,K=1,3)
```

```
      T1T=J1-J2,J1=1
```

```
      J1=1,J2=NK,J=1,1
```

```
      1. WRITE OUTPUT TAPE $1,102
```

```
      CALL EXIT
```

```
      11. CONTINUE
```

```
      DO IT=2,TOTALBAR
```

```
      READ INPUT TAPE $1,102,J,NDMP,(X11,J1,J=1,1)
```

```
      T1P=J1-J2,J1=1
```

```
      READ INPUT TAPE $1,103,J,NDMP,(X11,J1,J=1,1)
```

```
      12. CONTINUE
```

```
      DO IT=TOTALBAR
```

```
      READ INPUT TAPE $1,102,J,NDMP,(X11,J1,J=1,1)
```

```
      T1P=J1-J2,J1=1
```

```
      READ INPUT TAPE $1,103,J,NDMP,(X11,J1,J=1,1)
```

```
      13. CONTINUE
```

```
      NM=NJ+NEDOF,J
```

```
      NDMP=IT-NK
```

```
      NK=NK-NM
```

```
      NK=1
```

```
      IF(J1>1)THEN
```

```
      DO 30 K=1,1,NDF,J
```

```
      DO 30 XY=1,NDMP,J
```

```
      30 ALKX,KX=1,1
```

```
      20 ALKX,KX=1,1
```

```
      IF(J1>1)THEN
```

```
      DO 30 K=1,1,NDF,J
```

```
      DO 30 XY=1,NDMP,J
```

```
      30 MYX,YX=1,1
```

```
      22 TPTJBTCRBT=RCT25723725
```

```
      23 KX=JTA(K1)
```

```
      24 CALL STRTICKE(RX,KX)
```

```
      25 CONTINUE
```

```
      DO 26 K=1,NDF,J
```

```
      DO 26 XY=1,NDMP,J
```

```
      26 XY=1,NDMP,J
```

```
      27 ALKX,XY=1,NDMP,J
```

```
      28 CNTINUE
```

```
      DO 32 KX=1,NDF,J
```

```
      IF(LDEL(KD)-M130>32)30
```

```
      DO 32 KX=1,NDF,J
```

```
      30 CNTINUE
```

```
      DO 31 KY=1,NK
```

```
      31 XY=1,NDMP,J
```

```
      XY=1,NDMP,J
```

```
      DO 29 J=1,NL
```

```
      DO 29 J=1,NL
```

```
      29 ALI,J=1,0
```

```
      C EDIT AND OUTPUT STIFFNESS MATRIX
```

```
      C
```

```
      1F1NEDIT163163>61
```

```
      61 DO 62 K=NEDOF
```

```
      READ INPUT TAPE $1,101,J,NJK,NDE,J
```

```
      WRITE TAPE 1,((ATK(J,1),J,1),NK),(1,1,NK),1,1,CFS
```

```
      62 ATK(J,NJK)=1,J=NDE+1
```

```
      63 CY=0,0
```

```
      DO 33 T1=1,NK
```

```
      33 CKSCKS(KT,J,1)
```

```
      BEEND,J=1,NK
```

```
      WRITE TAPE 1,((ATK(J,1),J,1),NK),(1,1,NK),1,1,CFS
```

```
      JF1NBUFL12335,36
```

```
      WRITE OUTPUT TAPE $0,T0,T0P
```

```
      36 WRITE OUTPUT TAPE $0,101,J,NDMP,(X11,J1,J=1,1,NK)
```

```
      C READ AND GENERATE MASS AND LOAD MATRICES
```

```
      C
```

```
      35 K=1,J=1,1,NDF,J
```

```
      DO 36 K=1,J=1,1,NDF,J
```

```
      36 XY=1,NDMP,J
```

```
      36 XY=1,NDMP,J
```

```
      40 READ INPUT TAPE $0,NDMP,(X11,J1,J=1,1,NDF,J)
```

```
      41 KX=1,1,NDF,J
```

```
      42 MY=1,1,NDF,J
```

```
      DO 43 J=1,NDF,J
```

```
      ALKX,KX=1,1
```

```
      43 MYX,YX=1,1
```

```
      44 TPTJBTCRBT=RCT25723725
```









DO 13 KB=1NBAR  
IF (KINDA=1) AL1=1,0

10 E=A(KE0,0)

11 J=1,N,1

12 J=1,N,1

13 J=1,N,1

14 J=1,N,1

15 J=1,N,1

16 J=1,N,1

17 J=1,N,1

18 J=1,N,1

19 J=1,N,1

20 J=1,N,1

21 J=1,N,1

22 J=1,N,1

23 J=1,N,1

24 J=1,N,1

25 J=1,N,1

26 J=1,N,1

27 J=1,N,1

28 J=1,N,1

29 J=1,N,1

30 J=1,N,1

31 J=1,N,1

32 J=1,N,1

33 J=1,N,1

34 J=1,N,1

35 J=1,N,1

36 J=1,N,1

37 J=1,N,1

38 J=1,N,1

39 J=1,N,1

40 J=1,N,1

41 J=1,N,1

42 J=1,N,1

43 J=1,N,1

44 J=1,N,1

45 J=1,N,1

46 J=1,N,1

47 J=1,N,1

48 J=1,N,1

49 J=1,N,1

50 J=1,N,1

51 J=1,N,1

52 J=1,N,1

53 J=1,N,1

54 J=1,N,1

55 J=1,N,1

56 J=1,N,1

57 J=1,N,1

58 J=1,N,1

59 J=1,N,1

60 J=1,N,1

61 J=1,N,1

62 J=1,N,1

63 J=1,N,1

64 J=1,N,1

65 J=1,N,1

66 J=1,N,1

67 J=1,N,1

68 J=1,N,1

69 J=1,N,1

70 J=1,N,1

71 J=1,N,1

72 J=1,N,1

73 J=1,N,1

74 J=1,N,1

75 J=1,N,1

76 J=1,N,1

77 J=1,N,1

78 J=1,N,1

79 J=1,N,1

80 J=1,N,1

81 J=1,N,1

82 J=1,N,1

83 J=1,N,1

84 J=1,N,1

85 J=1,N,1

86 J=1,N,1

87 J=1,N,1

88 J=1,N,1

89 J=1,N,1

90 J=1,N,1

91 J=1,N,1

92 J=1,N,1

93 J=1,N,1

94 J=1,N,1

95 J=1,N,1

96 J=1,N,1

97 J=1,N,1

98 J=1,N,1



```

      SUBROUTINE R1V1R1AL
      45. WRITE OUTPUT TAPE 6+102+IV,IV,IV,IV,IV,IV
      C  SUBROUTINE FOR MEDIUM
      C
      C  31. CALL RICH 16,IV
      C
      R1V1R1AL+B
      IF 16 = C61 46457
      7 ABAB+B,0.5
      6 .9 + R1V1*(ABAB+0.5)
      A WRITE OUTPUT TAPE 6+102+IV,IV,IV,IV,IV,IV
      C STORE MODE SHAPES
      C
      15 DO 8 IV=1,N
      KVN+N IV
      DO 12 J=1,N
      12 V1V1TVCT(V1V1TVCT)
      CALL RICH
      DO 8 IV=1,N
      8 AX11(K)V1V1,1
      C COMPUTE HIGH EIGENVALUE OF A
      C
      24 R1MDO 16
      READING TAPE 6+102+IV,IV,IV,IV,IV,IV
      DO 9 IV=1,N
      9 V1V1TVCT
      IV,IV+
      DO 11 TR1TR1C
      ABAB+
      DO 10 TR1TR1C
      ABAB+
      10 V1V1TVCT
      9 + AB1IV)
      CALL RICH 16,IV
      B1V1R1AL+B
      IFEE-B,0.00028616110
      7: CONTINUE
      9 AB1IV)
      11 CALL RICH 16,IV
      61 RMAX = RAB+B,MM,MM,V1V1TVCT
      WRITE OUTPUT TAPE 6+102+IV,IV,IV,IV,IV,IV
      C COMPUTE LOW EIGENVALUES OF A
      C
      DO 13 IV=1,N
      JV,IV+N
      DO 14 IV=1,N
      14 V1211IVCIV,1
      BARAK
      CALL RICH 16,IV
      R1V1R1AL+B
      3=R1V1R1AL+B,MM
      WRITE OUTPUT TAPE 6+102+IV,IV,IV,IV,IV,IV
      CALL RICH 16,IV
      XE1C
      DO 42 I=1,165
      42 ABAB=B,12
      C ELIMINATE FORWARD AND BACKWARD
      C
      JV 15 = 16,4,5,6

```

```

6 DO 7 J=1,N
  15 J=I,J=AK(I,J),AK(I,K),I=AK(I,J)
7 CONTINUE
C   CENTER WITHIN TERRANE
C   PERFORMS ONE VECTOR ITERATION
  SUBROUTINE RICH TB(PT)
  COMMON N,ML,NM,VA,VC,AM,JA,JI,BA,X,MMOD,EYNG,POIS,
  XN1,NAR,NDF,IND,ICLDE,PIN,VAL,IV,
  DIMENSION AK(130,136),VC(18,135),AM(180),JA(200),
  PT(128,32),X780,31,DEL100,C,PT(127),PT(127),PT(130)
  CALL MDC(TB)
  IF(LT1,L1)1,2
  1 10  I=1,N
    11 1=I,I+1
    12  V=VAL(I,1)
    13  VAL(I,1)=V
    14  RETURN
  2 10  I=1,N
    11 1=I,I+1
    12  V=VAL(I,1)
    13  VAL(I,1)=V
    14  RETURN
END

```

```

C   ORTHOGONALIZE ONE VECTOR TO ANOTHER
  C   SUBROUTINE ORTH( L1,N1 )
  COMMON N,ML,NM,VA,VC,AM,JA,JI,BA,X,MMOD,EYNG,POIS,
  XN1,NAR,NDF,IND,ICLDE,PIN,VAL,IV,
  DIMENSION AK(130,136),VC(18,135),AM(180),JA(200),
  VA(200),A(160,31),DEL1(100),R(121),V(3,130)
  V(1)=0
  DO 1 I=1,N
    UN=MVAL(V(1),V(1))
    1  V=VAL(V(1),V(1))
    2  VAL(V(1),V(1))=0
    3  RAL=VN/VD
    GO TO 134,140
  C   DIRECTION 4 ORTHOGONALIZE V1,V2,V3 TO -V1,V2,V3
  4  DO 5 I=1,N
    5  V(1)=VAL(V(1))-RAL*V(IV,1)
  C   DIRECTION 3 RETURNS WITH RALEIGH QUOTIENT
  6  RETURN
END
C   FIND LARGEST ABSOLUTE CHANGE IN VECTOR COMPONENT
  C   SUBROUTINE EHR
  COMMON N,ML,NM,VA,VC,AM,JA,JI,BA,X,MMOD,EYNG,POIS,
  XN1,NAR,NDF,IND,ICLDE,PIN,VAL,IV,
  DIMENSION K(130,136),VAL(135),AM(180),JA(200),
  VA(200),A(160,31),DEL1(100),R(121),V(3,130),
  ER(131)
  PC 1=L1,N
  ETAG=FLASF(V1,V2,V3)-ABSF(V12,V2,V3)
  1 10  E=ETAG
    11  LE=L-E/LE
    12  IF(LE.LE.0.0)GO TO 14
    13  CONTINUE
    14  RETURN
END
C   NORMALIZE VECTOR TO VNORM
  C   SUBROUTINE NORM
  COMMON N,ML,NM,VA,VC,AM,JA,JI,BA,X,MMOD,EYNG,POIS,
  XN1,NAR,NDF,IND,ICLDE,PIN,VAL,IV
  DIMENSION AK(130,136),VC(18,135),AM(180),JA(200),
  VA(200),A(160,31),DEL1(100),R(121),V(3,130)
  TEST=0.0
  DO 1 I=1,N
    1 10  J=1,N
      11  V1=VAL(I,1)
      12  V2=VAL(I,2)
      13  V3=VAL(I,3)
      14  TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    15  J=J+1
    16  V1=VAL(I,1)
    17  V2=VAL(I,2)
    18  V3=VAL(I,3)
    19  TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    20  J=J+1
    21  V1=VAL(I,1)
    22  V2=VAL(I,2)
    23  V3=VAL(I,3)
    24  TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    25  J=J+1
    26  V1=VAL(I,1)
    27  V2=VAL(I,2)
    28  V3=VAL(I,3)
    29  TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    30  J=J+1
    31  V1=VAL(I,1)
    32  V2=VAL(I,2)
    33  V3=VAL(I,3)
    34  TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    35  J=J+1
    36  V1=VAL(I,1)
    37  V2=VAL(I,2)
    38  V3=VAL(I,3)
    39  TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    40  J=J+1
    41  V1=VAL(I,1)
    42  V2=VAL(I,2)
    43  V3=VAL(I,3)
    44  TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    45  J=J+1
    46  V1=VAL(I,1)
    47  V2=VAL(I,2)
    48  V3=VAL(I,3)
    49  TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    50  J=J+1
    51  V1=VAL(I,1)
    52  V2=VAL(I,2)
    53  V3=VAL(I,3)
    54  TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    55  J=J+1
    56  V1=VAL(I,1)
    57  V2=VAL(I,2)
    58  V3=VAL(I,3)
    59  TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    60  J=J+1
    61  V1=VAL(I,1)
    62  V2=VAL(I,2)
    63  V3=VAL(I,3)
    64  TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    65  J=J+1
    66  V1=VAL(I,1)
    67  V2=VAL(I,2)
    68  V3=VAL(I,3)
    69  TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    70  J=J+1
    71  V1=VAL(I,1)
    72  V2=VAL(I,2)
    73  V3=VAL(I,3)
    74  TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    75  J=J+1
    76  V1=VAL(I,1)
    77  V2=VAL(I,2)
    78  V3=VAL(I,3)
    79  TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    80  J=J+1
    81  V1=VAL(I,1)
    82  V2=VAL(I,2)
    83  V3=VAL(I,3)
    84  TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    85  J=J+1
    86  V1=VAL(I,1)
    87  V2=VAL(I,2)
    88  V3=VAL(I,3)
    89  TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    90  J=J+1
    91  V1=VAL(I,1)
    92  V2=VAL(I,2)
    93  V3=VAL(I,3)
    94  TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    95  J=J+1
    96  V1=VAL(I,1)
    97  V2=VAL(I,2)
    98  V3=VAL(I,3)
    99  TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    100 J=J+1
    101 V1=VAL(I,1)
    102 V2=VAL(I,2)
    103 V3=VAL(I,3)
    104 TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    105 J=J+1
    106 V1=VAL(I,1)
    107 V2=VAL(I,2)
    108 V3=VAL(I,3)
    109 TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    110 J=J+1
    111 V1=VAL(I,1)
    112 V2=VAL(I,2)
    113 V3=VAL(I,3)
    114 TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    115 J=J+1
    116 V1=VAL(I,1)
    117 V2=VAL(I,2)
    118 V3=VAL(I,3)
    119 TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    120 J=J+1
    121 V1=VAL(I,1)
    122 V2=VAL(I,2)
    123 V3=VAL(I,3)
    124 TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    125 J=J+1
    126 V1=VAL(I,1)
    127 V2=VAL(I,2)
    128 V3=VAL(I,3)
    129 TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    130 J=J+1
    131 V1=VAL(I,1)
    132 V2=VAL(I,2)
    133 V3=VAL(I,3)
    134 TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    135 J=J+1
    136 V1=VAL(I,1)
    137 V2=VAL(I,2)
    138 V3=VAL(I,3)
    139 TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    140 J=J+1
    141 V1=VAL(I,1)
    142 V2=VAL(I,2)
    143 V3=VAL(I,3)
    144 TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    145 J=J+1
    146 V1=VAL(I,1)
    147 V2=VAL(I,2)
    148 V3=VAL(I,3)
    149 TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    150 J=J+1
    151 V1=VAL(I,1)
    152 V2=VAL(I,2)
    153 V3=VAL(I,3)
    154 TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    155 J=J+1
    156 V1=VAL(I,1)
    157 V2=VAL(I,2)
    158 V3=VAL(I,3)
    159 TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    160 J=J+1
    161 V1=VAL(I,1)
    162 V2=VAL(I,2)
    163 V3=VAL(I,3)
    164 TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    165 J=J+1
    166 V1=VAL(I,1)
    167 V2=VAL(I,2)
    168 V3=VAL(I,3)
    169 TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    170 J=J+1
    171 V1=VAL(I,1)
    172 V2=VAL(I,2)
    173 V3=VAL(I,3)
    174 TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    175 J=J+1
    176 V1=VAL(I,1)
    177 V2=VAL(I,2)
    178 V3=VAL(I,3)
    179 TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    180 J=J+1
    181 V1=VAL(I,1)
    182 V2=VAL(I,2)
    183 V3=VAL(I,3)
    184 TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    185 J=J+1
    186 V1=VAL(I,1)
    187 V2=VAL(I,2)
    188 V3=VAL(I,3)
    189 TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    190 J=J+1
    191 V1=VAL(I,1)
    192 V2=VAL(I,2)
    193 V3=VAL(I,3)
    194 TEST=TEST+(V1*V1)+(V2*V2)+(V3*V3)
    195 J=J+1
    196 V1=VAL(I,1)
    197 V2=VAL(I,2)
    198 V3=VAL(I,3)
    199 TEST=TEST-(V1*V1)-(V2*V2)-(V3*V3)
    200 J=J+1
  END

```

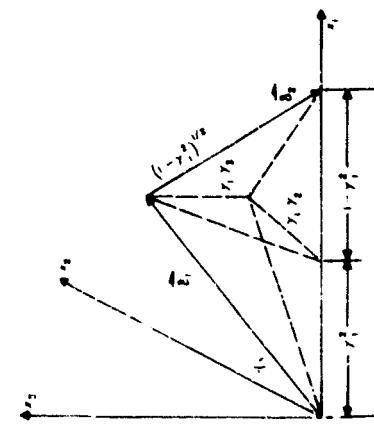
$$f_{11} = -\frac{AE}{S} \gamma_1 \gamma_1$$

$$f_{12} = -\frac{AE}{S} \gamma_1 \gamma_2$$

The matrix relating displacements of  $p$  to forces at  $p$  and  $q$  is

$$\begin{bmatrix} \gamma_1^2 & \gamma_1 \gamma_2 & \gamma_1 \gamma_3 \\ \gamma_1 \gamma_2 & \gamma_2^2 & \gamma_2 \gamma_3 \\ \gamma_1 \gamma_3 & \gamma_2 \gamma_3 & \gamma_3^2 \end{bmatrix}$$

### 2. Right-Jointed Member, Circular Cross Section, Three Dimensions (Sketch A-2)



Sketch A-2

Introduce  $\gamma_1 = 1$ . Vector displacements in the axial and transverse directions are

$$\begin{aligned} \vec{\delta}_1 &= \gamma_1^2 \vec{s}_1 + \gamma_1 \gamma_2 \vec{s}_2 + \gamma_1 \gamma_3 \vec{s}_3 \\ \vec{\delta}_2 &= (1 - \gamma_1^2) \vec{s}_1 - \gamma_1 \gamma_2 \vec{s}_2 - \gamma_1 \gamma_3 \vec{s}_3. \end{aligned}$$

A unit vector normal to  $\vec{\delta}_1$  and  $\vec{\delta}_2$  is

$$\begin{aligned} \vec{n} &= \frac{\vec{\delta}_1 \times \vec{\delta}_2}{|\vec{\delta}_1| |\vec{\delta}_2|} \\ &= \frac{(2\gamma_1 \gamma_2 \vec{s}_3 - 2\gamma_1^2 \vec{s}_2)}{\sqrt{\gamma_1^4(1-\gamma_1^2)^2}} \\ &= \frac{(2\gamma_1 \vec{s}_3 - 2\gamma_1^2 \vec{s}_2)}{(1-\gamma_1^2)^{3/2}}. \end{aligned}$$

The vector force exerted on

$$p = \frac{AE}{S} \vec{s}_1 + \frac{12 EI}{S^3} \vec{n},$$

and the vector moment at

$$p = \frac{6 EI}{S} (1 - \gamma_1^2) \vec{s}_3.$$

Components of these load vectors are

$$f_{11} = \text{force along } x_1 \text{ axis} = -\frac{AE}{S} \gamma_1 + \frac{12 EI}{S^3} (1 - \gamma_1^2)$$

$$f_{12} = \text{force along } x_2 \text{ axis} = \left(\frac{AE}{S} - \frac{12 EI}{S^3}\right) \gamma_1 \gamma_2$$

$$f_{13} = \text{force along } x_3 \text{ axis} = \left(\frac{AE}{S} - \frac{12 EI}{S^3}\right) \gamma_1 \gamma_3$$

$$f_{21} = \text{moment about } x_1 \text{ axis} = 0$$

$$f_{22} = \text{moment about } x_2 \text{ axis} = \frac{6 EI}{S^3} \gamma_1$$

$$f_{23} = \text{moment about } x_3 \text{ axis} = \frac{6 EI}{S^3} \gamma_1$$

Similar load components at  $q$  are

$$f_{31} = -\frac{AE}{S} \gamma_1^2 + \frac{12 EI}{S^3} (1 - \gamma_1^2)$$

$$f_{32} = \left(\frac{AE}{S} + \frac{12 EI}{S^3}\right) \gamma_1 \gamma_2$$

$$f_{33} = \left(\frac{AE}{S} + \frac{12 EI}{S^3}\right) \gamma_1 \gamma_3$$

Section properties:

$$D \neq A,$$

$$T = A,$$

$$A = T(D - T),$$

$$I_A := 0$$

$$I_{Ax} := \frac{6 EI}{S^3} \gamma_1$$

$$f_{11} = \frac{6 EI}{S^3} \gamma_1$$

The required matrix will be written in terms of the quantities

$$C_0 = \frac{AE}{S}$$

$$C_1 = \frac{EI}{S^3(1+\sigma)}$$

$$C_{12} = \frac{12 EI}{S^3}$$

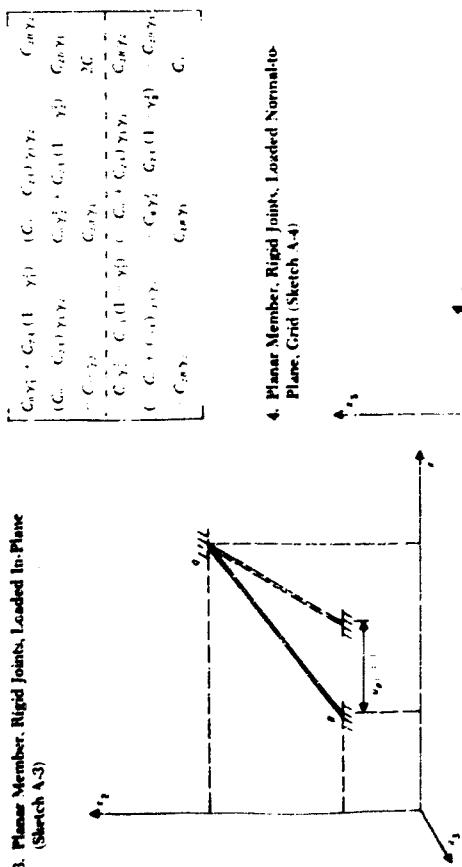
$$C_{13} = \frac{6 EI}{S^3}$$

$$C_2 = \frac{6 EI}{S}$$

$$C_3 = \frac{2 EI}{S}$$

$$\boxed{\begin{bmatrix} C_0 \gamma_1^2 & (C_0 - C_{11}) \gamma_1 \gamma_2 & (C_0 - C_{11}) \gamma_1 \gamma_3 & C_{11} \gamma_1 \\ (C_0 - C_{11}) \gamma_1 \gamma_2 & C_0 \gamma_2^2 & (C_0 - C_{11}) \gamma_2 \gamma_3 & C_{11} \gamma_2 \\ (C_0 - C_{11}) \gamma_1 \gamma_3 & (C_0 - C_{11}) \gamma_2 \gamma_3 & C_0 \gamma_3^2 & C_{11} \gamma_3 \\ C_{11} \gamma_1 & C_{11} \gamma_2 & C_{11} \gamma_3 & -C_{11} \gamma_1 \\ C_{12} \gamma_1^2 & (C_0 + C_{12}) \gamma_1 \gamma_2 & (C_0 + C_{12}) \gamma_1 \gamma_3 & C_{12} \gamma_1 \\ (C_0 + C_{12}) \gamma_1 \gamma_2 & C_0 \gamma_2^2 & (C_0 + C_{12}) \gamma_2 \gamma_3 & C_{12} \gamma_2 \\ (C_0 + C_{12}) \gamma_1 \gamma_3 & (C_0 + C_{12}) \gamma_2 \gamma_3 & C_0 \gamma_3^2 & C_{12} \gamma_3 \\ C_{12} \gamma_2^2 & C_{12} \gamma_1 \gamma_2 & (C_0 + C_{12}) \gamma_1 \gamma_3 & C_{12} \gamma_2 \\ C_{12} \gamma_1 \gamma_3 & (C_0 + C_{12}) \gamma_2 \gamma_3 & C_0 \gamma_3^2 & C_{12} \gamma_3 \\ C_{13} \gamma_1^2 & (C_0 + C_{13}) \gamma_1 \gamma_2 & (C_0 + C_{13}) \gamma_1 \gamma_3 & C_{13} \gamma_1 \\ (C_0 + C_{13}) \gamma_1 \gamma_2 & C_0 \gamma_2^2 & (C_0 + C_{13}) \gamma_2 \gamma_3 & C_{13} \gamma_2 \\ (C_0 + C_{13}) \gamma_1 \gamma_3 & (C_0 + C_{13}) \gamma_2 \gamma_3 & C_0 \gamma_3^2 & C_{13} \gamma_3 \\ C_{13} \gamma_2^2 & C_{13} \gamma_1 \gamma_2 & (C_0 + C_{13}) \gamma_1 \gamma_3 & C_{13} \gamma_2 \\ C_{13} \gamma_1 \gamma_3 & (C_0 + C_{13}) \gamma_2 \gamma_3 & C_0 \gamma_3^2 & C_{13} \gamma_3 \\ C_{21} \gamma_1^2 & (C_0 - C_{21}) \gamma_1 \gamma_2 & (C_0 - C_{21}) \gamma_1 \gamma_3 & C_{21} \gamma_1 \\ (C_0 - C_{21}) \gamma_1 \gamma_2 & C_0 \gamma_2^2 & (C_0 - C_{21}) \gamma_2 \gamma_3 & C_{21} \gamma_2 \\ (C_0 - C_{21}) \gamma_1 \gamma_3 & (C_0 - C_{21}) \gamma_2 \gamma_3 & C_0 \gamma_3^2 & C_{21} \gamma_3 \\ C_{21} \gamma_2^2 & C_{21} \gamma_1 \gamma_2 & (C_0 - C_{21}) \gamma_1 \gamma_3 & C_{21} \gamma_2 \\ C_{21} \gamma_1 \gamma_3 & (C_0 - C_{21}) \gamma_2 \gamma_3 & C_0 \gamma_3^2 & C_{21} \gamma_3 \\ C_{31} \gamma_1^2 & (C_0 - C_{31}) \gamma_1 \gamma_2 & (C_0 - C_{31}) \gamma_1 \gamma_3 & C_{31} \gamma_1 \\ (C_0 - C_{31}) \gamma_1 \gamma_2 & C_0 \gamma_2^2 & (C_0 - C_{31}) \gamma_2 \gamma_3 & C_{31} \gamma_2 \\ (C_0 - C_{31}) \gamma_1 \gamma_3 & (C_0 - C_{31}) \gamma_2 \gamma_3 & C_0 \gamma_3^2 & C_{31} \gamma_3 \\ C_{31} \gamma_2^2 & C_{31} \gamma_1 \gamma_2 & (C_0 - C_{31}) \gamma_1 \gamma_3 & C_{31} \gamma_2 \\ C_{31} \gamma_1 \gamma_3 & (C_0 - C_{31}) \gamma_2 \gamma_3 & C_0 \gamma_3^2 & C_{31} \gamma_3 \end{bmatrix}}$$

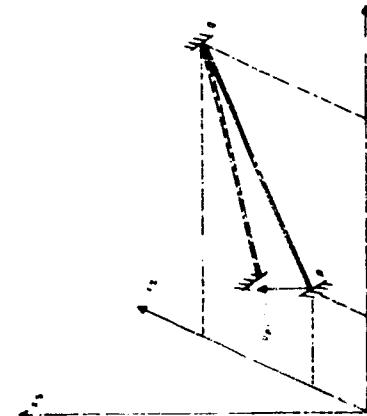
**3. Planar Member, Rigid Joints, Loaded In-Plane  
(Sketch A-3)**



Introduce  $\theta_1$ , i.e. Moment about an axis transverse to the member is of magnitude  $6 EB/S^2$ . Components of load exerted on  $p$  and  $q$  are

$$\begin{aligned} f_{px} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{py} &= C_{11} \gamma_1 \\ f_{qx} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{qy} &= C_{12} \gamma_1 \\ f_{rx} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{ry} &= C_{12} \gamma_1 \\ f_{sx} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{sy} &= C_{12} \gamma_1 \\ f_{tx} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{ty} &= C_{12} \gamma_1 \\ f_{ux} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{uy} &= C_{12} \gamma_1 \\ f_{vx} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{vy} &= C_{12} \gamma_1 \\ f_{wx} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{wy} &= C_{12} \gamma_1 \end{aligned}$$

**4. Planar Member, Rigid Joints, Loaded Normal to Plane, Grid (Sketch A-4)**



**Sketch A-4**

**Section properties:**

$$\begin{aligned} A &= A_1 \\ I &= A_1 \\ D &= A_1 \\ T &= A_1 \end{aligned}$$

If  $A_1 = 0$  compute

$$A = D_1 \\ I = \frac{\sum T}{8}$$

**Section properties:**

$$\begin{aligned} B &= A_1 \\ K &= A_1 \\ W &= A_1 \\ H &= A_1 \end{aligned}$$

The derivation is similar to that preceding with  $y_1 = 0$ . The  $\vec{r}_1$  is written in terms of

$$\begin{aligned} C_{11} &= \frac{4F}{S} & C_{12} &= \frac{6EI}{S^2} \\ C_{11} &= \frac{12EI}{S^2} & C_{12} &= \frac{2EI}{S} \end{aligned}$$

if  $A_1 = 0$ , compute

$$\begin{aligned} B &= \frac{W H}{12(1 - \nu^2)} \\ K &= \frac{W H}{\nu^2} \\ H &= A_1 \end{aligned}$$

Loads are in the order

- $f_{px}$  = force along  $x_1$  axis
- $f_{py}$  = force along  $y_1$  axis
- $f_{rz}$  = moment about  $x_1$  axis

Introduce  $\theta_1$ , i.e. Moment about an axis transverse to the member is of magnitude  $6 EB/S^2$ . Components of load exerted on  $p$  and  $q$  are

$$\begin{aligned} f_{px} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{py} &= C_{11} \gamma_1 \\ f_{qx} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{qy} &= C_{12} \gamma_1 \\ f_{rx} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{ry} &= C_{12} \gamma_1 \\ f_{sx} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{sy} &= C_{12} \gamma_1 \\ f_{tx} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{ty} &= C_{12} \gamma_1 \\ f_{ux} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{uy} &= C_{12} \gamma_1 \\ f_{vx} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{vy} &= C_{12} \gamma_1 \\ f_{wx} &= C_{11} \gamma_1 + C_{12} (1 - \gamma_1) & f_{wy} &= C_{12} \gamma_1 \end{aligned}$$

$$\begin{aligned} C_{11} &= \frac{4F}{S} & C_{12} &= \frac{6EI}{S^2} \\ C_{11} &= \frac{12EI}{S^2} & C_{12} &= \frac{2EI}{S} \\ C_{11} &= \frac{6EB}{S^2} & C_{12} &= \frac{2EB}{S} \\ C_{11} &= \frac{2EB}{S^2} & C_{12} &= \frac{C_1 \gamma_1}{S^2} \\ C_{11} &= \frac{C_1 \gamma_1}{S^2} & C_{12} &= \frac{(C_1 - 6C_1 \gamma_1) \gamma_1}{S^2} \\ C_{11} &= \frac{C_1 \gamma_1}{S^2} & C_{12} &= \frac{(C_1 - 2C_1 \gamma_1) \gamma_1}{S^2} \\ C_{11} &= \frac{C_1 \gamma_1}{S^2} & C_{12} &= \frac{C_1 \gamma_1}{S^2} \\ C_{11} &= \frac{C_1 \gamma_1}{S^2} & C_{12} &= \frac{(C_1 - 2C_1 \gamma_1) \gamma_1}{S^2} \\ C_{11} &= \frac{C_1 \gamma_1}{S^2} & C_{12} &= \frac{(C_1 - 6C_1 \gamma_1) \gamma_1}{S^2} \\ C_{11} &= \frac{C_1 \gamma_1}{S^2} & C_{12} &= \frac{(C_1 - C_1 \gamma_1) \gamma_1}{S^2} \\ C_{11} &= \frac{C_1 \gamma_1}{S^2} & C_{12} &= \frac{(C_1 - C_1 \gamma_1) \gamma_1}{S^2} \end{aligned}$$

As before, the matrix is written in terms of the parameters

## APPENDIX B

### Stresses for Various Member Types

Expressions for member stresses are developed, using the geometrical parameters of Section III, and joint deflections in the order specified in Section III-A4.

#### 1. Pin-Jointed Member, Three Dimensions

Atrial extension of the member is

$$\delta = (\mathbf{u}_{p_1} - \mathbf{u}_{p_2}) \cdot \gamma + (\mathbf{u}_{p_1} - \mathbf{u}_{p_2}) \cdot \gamma + (\mathbf{u}_{p_1} - \mathbf{u}_{p_2}) \cdot \gamma$$

The axial stress is computed and output for each loading on each member in pound units, tension positive:

$$\sigma_p = \frac{AE}{\delta} \cdot \delta_{p_1}$$

#### 2. Rigid-Jointed Member, Circular Cross Section, Three Dimensions

A member-oriented coordinate system is defined as follows:

$$\begin{aligned} \vec{\xi}_1 & \text{ unit vector along member axis} \\ \vec{\gamma}_1 & \vec{\xi}_1 \times \vec{\gamma}_2 \quad \vec{\gamma}_2 \text{ unit vector normal to plane of } \vec{\xi}_1 \text{ and } \vec{x} \\ \vec{\gamma}_3 & \vec{\xi}_1 \times \vec{\gamma}_2 \quad \vec{\gamma}_3 \text{ unit vector normal to plane of } \vec{\xi}_1 \text{ and } \vec{x} \\ & (\text{in } x_1, \text{ if } \vec{\xi}_1 \perp \vec{\gamma}_1) \end{aligned}$$

$$\begin{aligned} \vec{\xi}_1 & \frac{\vec{\xi}_1}{\|\vec{\xi}_1\|}, \quad \vec{\gamma}_2 \frac{\vec{\gamma}_2}{\|\vec{\gamma}_2\|}, \quad \vec{\gamma}_3 \frac{\vec{\gamma}_3}{\|\vec{\gamma}_3\|} \\ \vec{\xi}_1 & \vec{\xi}_1 \times \vec{\gamma}_2 \end{aligned}$$

$$\begin{aligned} \vec{\xi}_1 & \text{ unit vector normal to plane of } \vec{\xi}_1 \text{ and } \vec{\xi}_2 \\ \vec{\xi}_1 & \vec{\xi}_1 \times \vec{\xi}_2 \\ \vec{\gamma}_1 & \vec{\xi}_1 \times \vec{\gamma}_2 \quad \vec{\gamma}_2 \text{ unit vector normal to plane of } \vec{\gamma}_1 \text{ and } \vec{x} \\ \vec{\gamma}_3 & \vec{\gamma}_1 \times \vec{\gamma}_2 \quad \vec{\gamma}_3 \text{ unit vector normal to plane of } \vec{\gamma}_1 \text{ and } \vec{x} \end{aligned}$$

The following quantities are output, in order, for each member:

$$\begin{aligned} P & \text{ axial stress} \\ & = \frac{AE}{\delta} \cdot \delta_{p_1} \end{aligned}$$

$M_p$  = resultant bending moment at  $p$

$$\begin{aligned} M_p & = (\mathbf{M}_{p_1} + \mathbf{M}_{p_2})^{\perp} \\ & = (M_{p_1} + M_{p_2})^{\perp} \end{aligned}$$

$M_t$  = resultant shear at  $p$

$$\begin{aligned} M_t & = (M_{p_1} + M_{p_2})^{\parallel} \\ & = (M_{p_1} + M_{p_2})^{\parallel} \end{aligned}$$

Net displacement components in the  $\vec{\xi}_1^{\perp}$  directions are

$$\delta_{p_1} = (\mathbf{u}_{p_1} - \mathbf{u}_{p_2}) \cdot \vec{\xi}_1^{\perp} + (\mathbf{u}_{p_1} - \mathbf{u}_{p_2}) \cdot \vec{\xi}_1^{\perp} + (\mathbf{u}_{p_1} - \mathbf{u}_{p_2}) \cdot \vec{\xi}_1^{\perp}$$

$$\delta_{p_2} = (\mathbf{u}_{p_2} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp} + (\mathbf{u}_{p_2} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp} + (\mathbf{u}_{p_2} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp}$$

$$\delta_{p_3} = (\mathbf{u}_{p_3} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp} + (\mathbf{u}_{p_3} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp} + (\mathbf{u}_{p_3} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp}$$

$$\delta_{p_4} = (\mathbf{u}_{p_4} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp} + (\mathbf{u}_{p_4} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp} + (\mathbf{u}_{p_4} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp}$$

$$\delta_{p_5} = (\mathbf{u}_{p_5} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp} + (\mathbf{u}_{p_5} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp} + (\mathbf{u}_{p_5} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp}$$

$$\delta_{p_6} = (\mathbf{u}_{p_6} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp} + (\mathbf{u}_{p_6} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp} + (\mathbf{u}_{p_6} - \mathbf{u}_{p_1}) \cdot \vec{\xi}_1^{\perp}$$

$$= \frac{1}{S} [ (M_p + M_q) + (M_p + M_q)^2 ]^{\frac{1}{2}}$$

### 3. Planar Member, Rigid Joints, Loaded In-Plane

Axial extension of the member is

$$\delta_x = (a_{p1} - a_{p2}) \gamma_1 + (a_{q1} - a_{q2}) \gamma_2$$

Net transverse deflection of member is

$$\delta_y = (a_{p1} - a_{p2}) \gamma_1 + (a_{q1} - a_{q2}) \gamma_2$$

The following quantities are output, in order, for each member.

$$M_p \quad \text{bending moment at } p \\ = \frac{2EI}{S} \left( 2a_{p1} + a_{q1} - \frac{3b_1}{S} \right)$$

$$M_q \quad \text{bending moment at } q \\ = \frac{2EI}{S} \left( 2a_{q1} + a_{p2} - \frac{3b_2}{S} \right)$$

$$V_t \quad \text{shear at } p$$

$$= -\frac{1}{S} (M_p + M_q)$$

$P$  : axial stress

$$= -\frac{AE}{S} \delta_x$$

$$4. \text{ Planar Member, Rigid Joints, Loaded Normal to-Plane (Grid)}$$

Net transverse displacement (normal-to-plane) is

$$\delta_z = a_{p1} - a_{p2}$$

Net axial rotation is

$$\delta_x = (a_{p1} - a_{p2}) \gamma_1 + (a_{q1} - a_{q2}) \gamma_2$$

Transverse rotations of the ends are

$$\delta_{p1} = a_{p1} \gamma_1 + a_{p2} \gamma_2$$

$$\delta_{p2} = a_{p2} \gamma_1 + a_{q1} \gamma_2$$

The following quantities are output, in order, for each member:

$$M_p \quad \text{bending moment at } p \\ = \frac{2EB}{S} \left( 2a_{p1} + a_{q1} - \frac{3b_1}{S} \right)$$

$$M_q \quad \text{bending moment at } q \\ = \frac{2EB}{S} \left( 2a_{q1} + a_{p2} - \frac{3b_2}{S} \right)$$

$$M_t \quad \text{twisting moment}$$

$$= \frac{EC}{2S(1+\nu)} \delta_z$$

$$V_t \quad \text{shear at } p \text{ (normal-to-plane)}$$

$P$  : axial stress

$$= -\frac{1}{S} (M_p + M_q)$$

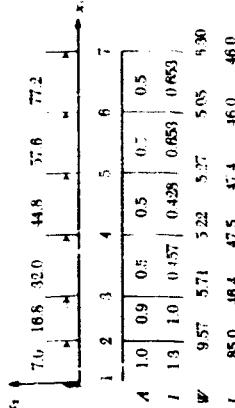
## APPENDIX C

### Example Problem

In the example problem contained in this Appendix, the cantilever span illustrated below will be analyzed for the following:

1. Static deflections and stresses for:

- a. 10 $g$  acceleration in the  $x$ , direction.
- b. 100-lb transverse force at tip
- c. Normal-mode shapes, frequencies, and dynamic stresses, assuming the maximum acceleration amplitudes to be 20 $g$  in the first three modes and 15 $g$  in the next three.



The material is aluminum. Section properties, weights,

and weight moments of inertia ( $I_w$ ), are specified

GO

**ACKNOWLEDGMENT**

The routines for solution of the static and normal-mode matrix equations were programmed by R. A. Rosanoff of the Programming Analysis Group, Jet Propulsion Laboratory.

**APPENDIX A**  
**Matrices for Various Member Types**
**1. Plan-Jointed Member, Three Dimensions (Sketch A-1)**

The following derivations are performed on typical members by introducing successive unit coordinate deflections of their ends and calculating forces reacting on the member. Coordinate deflections include both translations and rotations; loads are forces and moments. In each case, the first column of the required matrix is derived in some detail to illustrate procedure.

Matrices relating forces and displacements in structure-oriented ( $x_i$ ) coordinates are desired here, but intermediate use of member-oriented ( $t_i$ ) coordinates is made in the more complicated derivations. Inputs required for generation of the matrix of the most general member (arbitrary cross section in three dimensions) may be included in the present format, when the necessary program is written.

In the derivations below, the following quantities are input or computed for each member  $p = q$ :

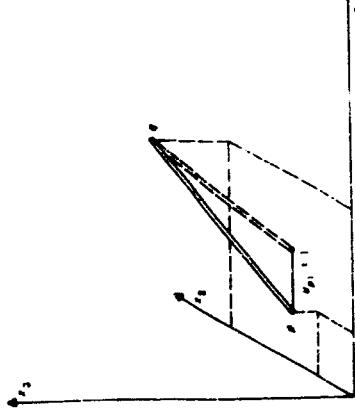
1. Input coordinate  $\alpha$ ,  $x_{pi}$ ,  $x_{qi}$
2. Input member properties,  $A_i$ ,  $E$
3. Compute member length
4. Compute direction cosines

$$\begin{aligned} S &= [(x_{q1} - x_{p1})^2 + (x_{q2} - x_{p2})^2 + (x_{q3} - x_{p3})^2]^{1/2} \\ \gamma_1 &= \frac{(x_{q1} - x_{p1})}{S} \\ \gamma_2 &= \frac{(x_{q2} - x_{p2})}{S} \\ \gamma_3 &= \frac{(x_{q3} - x_{p3})}{S} \end{aligned}$$

$$\begin{aligned} \text{Section property: } A &= A_i, \\ \text{Introduce: } s_{pi} &:= 1 \\ \text{Axial stress: } \sigma &= \frac{AE}{S} \gamma_1 \\ f_{pi} &= \frac{AE}{S} \gamma_1 \gamma_1 \\ f_{qi} &= \frac{AE}{S} \gamma_1 \gamma_2 \\ f_{pi} &= \frac{AE}{S} \gamma_1 \gamma_3 \\ f_{qi} &= \frac{AE}{S} \gamma_2 \gamma_3 \\ f_{pi} &= -\frac{AE}{S} \gamma_3 \end{aligned}$$

Matrices  $K_{pq}$ ,  $K_{qp}$  are written satisfying the expression

$$\begin{bmatrix} \vec{f}_p \\ \vec{f}_q \end{bmatrix} = \begin{bmatrix} K_{pq} \\ K_{qp} \end{bmatrix} \begin{bmatrix} \vec{s}_{pq} \\ \vec{s}_{qp} \end{bmatrix}$$

**Sketch A-1**







## BIBLIOGRAPHY

1. Crandall, S. H., *Engineering Analysis*, McGraw-Hill Book Co., Inc., New York, N. Y., 1956.
2. von Karman, T. and Biot, M. A., *Mathematical Methods in Engineering*, McGraw-Hill Book Co., Inc., New York, N. Y., 1940.
3. Frazer, R. A., Durbin, W. J., and Collar, A. R., *Elementary Matrices, and Some Applications to Dynamics and Differential Equations*, Cambridge University Press, New York, N. Y., 1955.
4. Lanczos, C., *Applied Analysis*, Prentice Hall Inc., Englewood Cliffs, N. J., 1956.
5. Sodewig, E., *Matrix Calculus*, Interscience Publishers Inc., New York, N. Y., 1956.
6. Norris, C. H., et al., *Structural Design for Dynamic Load*, McGraw-Hill Book Co., Inc., New York, N. Y., 1959.
7. Hidebrand, F. B., *Introduction to Numerical Analysis*, McGraw-Hill Book Co., Inc., New York, N. Y., 1956.
8. Scanlan, R. H. and Rosenbaum, R., *Introduction to the Study of Aircraft Vibration and Flutter*, The Macmillan Co., New York, N. Y., 1951.
9. Hall, S. A. and Woodhead, R. W., *Frame Analysis*, John Wiley and Sons, Inc., New York, N. Y., 1961.
10. Timoshenko, S., *Vibration Problems in Engineering*, D. Van Nostrand Co., Inc., New York, N. Y., 1955.
11. Roark, R. J., *Formulas for Stress and Strain*, McGraw-Hill Book Co., Inc., New York, N. Y., 1954, p. 174.
12. Argon, J. H., "Energy Theorems and Structural Analysis," *Aircraft Engineering*, October 1954, November 1954, February 1955, March 1955, April 1955, and May 1955.
13. Archer, J. S., "Digital Computation for Stiffness Matrix Analysis," *Proceedings of the American Society of Civil Engineers*, October 1958.
14. Archer, J. S. and Samon, C. H., Jr., "Structural Idealization for Digital Computer Analysis," *Conference Papers of the American Society of Civil Engineers*, Pittsburgh Conference on Electronic Computation, September 1958.
15. Martin, H. C., "Tors Analysis by Stiffness Consideration," *Proceedings of the American Society of Civil Engineers*, October 1956.
16. Clough, R. W., "Structural Analysis by Means of a Matrix Algebra Program," *Conference Papers of the American Society of Civil Engineers*, Kansas City Conference on Electronic Computation, November 1958.
17. Turner, M. J., Clough, R. W., Martin, H. C., and Topper, L. J., "Stiffness and Deflection Analysis of Complex Structures," *Journal of Aerospace Sciences*, September 1958.
18. Klein, D., "A Simple Method of Matrix Structural Analysis," *Journal of Aeronautical Sciences*, January 1957.
19. Martin, H. C., Jr. and Bergmann, H. W., "Analysis of Low Aspect Ratio Aircraft Structures," *Journal of Aerospace Sciences*, September 1960.
20. Melosh, R. J., "A Stiffness Matrix for the Analysis of Thin Plates in Bending," *Journal of Aerospace Sciences*, January 1961.
21. Martin, H. C., Jr. and Henaner, J., "Orthogonal Grids Worked Loaded Normally to Their Planes," *Journal of the American Society of Civil Engineers*, January 1960.
22. White, P. A., "The Computation of Eigenvalues and Eigenvectors of a Matrix," *Journal of the Society for Industrial and Applied Mathematics*, December 1958.

## BIBLIOGRAPHY

1. Crandall, S. H., *Engineering Analysis*, McGraw-Hill Book Co., Inc., New York, N. Y., 1955.
2. van Karman, T. and Biot, M. A., *Mathematical Methods in Engineering*, McGraw-Hill Book Co., Inc., New York, N. Y., 1940.
3. Frazer, R. A., Durbin, W. J., and Collar, A. R., *Elementary Matrices, and Some Applications to Dynamics and Differential Equations*, Cambridge University Press, New York, N. Y., 1955.
4. Lanczos, C., *Applied Analysis*, Prentice Hall Inc., Englewood Cliffs, N. J., 1956.
5. Sodergren, E., *Matrix Calculus*, Interscience Publishers Inc., New York, N. Y., 1956.
6. Norris, C. H., et al., *Structural Design for Dynamic Load*, McGraw-Hill Book Co., Inc., New York, N. Y., 1959.
7. Hildebrand, F. B., *Introduction to Numerical Analysis*, McGraw-Hill Book Co., Inc., New York, N. Y., 1956.
8. Scanlan, R. H. and Rosenbaum, R., *Introduction to the Study of Aircraft Vibration and Flutter*, The Macmillan Co., New York, N. Y., 1951.
9. Hull, S. A. and Woodhead, R. W., *Frame Analysis*, John Wiley and Sons, Inc., New York, N. Y., 1961.
10. Timoshenko, S., *Vibration Problems in Engineering*, D. Van Nostrand Co., Inc., New York, N. Y., 1955.
11. Roark, R. J., *Formulas for Stress and Strain*, McGraw-Hill Book Co., Inc., New York, N. Y., 1954, p. 174.
12. Argyris, J. H., *Energy Theorems and Structural Analysis*, Aircraft Engineering, October 1954, November 1954, February 1955, March 1955, April 1955, and May 1955.
13. Archer, J. S., "Digital Computation for Stiffness Matrix Analysis," *Proceedings of the American Society of Civil Engineers*, October 1958.
14. Archer, J. S. and Samson, C. H., Jr., "Structural Idealization for Digital Computer Analysis," *Conference Paper of the American Society of Civil Engineers*, Pittsburgh Conference on Electronic Computation, September 1958.
15. Martin, H. C., "Truss Analysis by Stiffness Considerations," *Proceedings of the American Society of Civil Engineers*, October 1956.
16. Clough, R. W., "Structural Analysis by Means of a Matrix Algebra Program," *Conference Paper of the American Society of Civil Engineers*, Kansas City, Conferene on Electronic Computation, November 1958.
17. Turner, M. J., Clough, R. W., Martin, H. C., and Todd, L. J., "Stiffness and Deflection Analysis of Complex Structures," *Journal of Aerospace Sciences*, September 1956.
18. Klein, D., "A Simple Method of Matrix Structural Analysis," *Journal of Aeronautical Sciences*, January 1958.
19. Samson, C. H., Jr. and Bergmann, H. W., "Analysis of Low-Aspect Ratio Aircraft Structures," *Journal of Aerospace Sciences*, September 1960.
20. Melosh, R. J., "A Stiffness Matrix for the Analysis of Thin Plates in Bending," *Journal of Aerospace Sciences*, January 1961.
21. Martin, I. and Hennann, J., "Orthogonal Grids," *Journal of the American Society of Civil Engineers*, January 1960.
22. White, P. A., "The Computation of Eigenvalues and Eigenvectors of a Matrix," *Journal of the Society for Industrial and Applied Mathematics*, December 1958.